Proceedings

THIRTY-FOURTH ANNUAL ALLERTON CONFERENCE ON COMMUNICATION, CONTROL, AND COMPUTING

October 2 — October 4, 1996

Allerton House, Monticello, Illinois
Sponsored by the
Coordinated Science Laboratory and the
Department of Electrical and Computer Engineering of the
University of Illinois at Urbana-Champaign

PROCEEDINGS

THIRTY-FOURTH ANNUAL ALLERTON CONFERENCE ON COMMUNICATION, CONTROL, AND COMPUTING

Sean P. Meyn
W. Kenneth Jenkins
Conference Co-Chairs

Conference held
October 2 — October 4, 1996
Allerton House
Monticello, Illinois

Sponsored by
The Coordinated Science Laboratory
and
The Department of Electrical and Computer Engineering
of the
UNIVERSITY OF ILLINOIS
at
Urbana-Champaign

621.3811
AL5
1996

This *Proceedings* contains papers presented at the Thirty-fourth Annual Allerton Conference on Communication, Control, and Computing which was held at Allerton House, Monticello, Illinois, on October 2, 3, and 4, 1996. As in past years, the Conference was sponsored by the Coordinated Science Laboratory and the Department of Electrical and Computer Engineering of the University of Illinois at Urbana-Champaign.

Sixty-four invited papers and fifty-three contributed papers were presented at the Conference in twenty-three sessions. We thank the authors for contributing to the success of the Conference. We are also very grateful to our colleagues at the University of Illinois at Urbana-Champaign: Professors B. Bamieh, R. Blahut, R. Braatz, B. Hajek, D. Jones, P. R. Kumar, U. Madhow, D. C. Munson, Jr., R. S. Sreenivas, R. Srikant, A. Vardy, and P. Voulgaris for organizing many of the invited sessions, reviewing the contributed papers, chairing sessions, and helping in many other ways in the smooth running of the Conference. Invited sessions were also chaired and in some cases also organized, by Professors V. Anantharam of the University of California, Berkeley, M. Jamshidi of the University of New Mexico, J. Sadowsky of Arizona State University, L. Tassiulas of the University of Maryland, G. A. Williamson of the Illinois Institute of Technology, K. Zeger of the University of California, San Diego. We are especially grateful to them for their efforts on behalf of the Conference.

The plenary speaker at the Conference was Dr. G. David Forney, Jr., Vice President of the Information Systems Group at Motorola, Inc., who gave a lecture entitled "The Viterbi Algorithm – Twenty-five Years Later: The Forward-Backward Algorithm." We are grateful to Dr. Forney for sharing his thoughts on this topic with us. His excellent presentation contributed much to the success of the Conference. We are also indebted to the "Pocket Big Band" for providing musical entertainment at the Conference.

We thank Mrs. Beverly Becker and her staff at Allerton House for cheerfully accommodating our Conference and for their constant attention to our needs. It is also a great pleasure to thank all the staff members of the Coordinated Science Laboratory who so ably handled all the practical aspects of organizing the Conference. In particular, we wish to mention Mr. Steven Williams and Ms. Susan Baxley who handled all the business affairs, and Mr. Dan Jordan who handled the shipping of these *Proceedings*. Most of all, however, we want to thank our Conference Secretary, Ms. Becky Lonberger, who with the able assistance of Ms. Nancy Carr and Ms. Francie Bridges worked all year round in handling announcements, calls for papers, program preparation, Allerton House arrangements, registration, and the final editing of these *Proceedings*. The lion's share of the credit for the success of the Conference belongs to these individuals.

W. Kenneth Jenkins and Sean P. Meyn

TABLE OF CONTENTS

ACKNOWLEDGEMENTS i

I-A: NETWORKS I
Organizer and Chairman: B. Hajek
(University of Illinois at Urbana-Champaign)

A CONTROL POLICY FOR THE STABILIZATION OF END-TO-END CONGESTION: MULTIPLE SOURCES CASE 1
H. Zhang, O.W. Yang, and H. Mouftah

SERVICE GUARANTEES FOR WINDOW FLOW CONTROL 10
R.L. Cruz and C.M. Okino

FAST ALGORITHMS FOR MEASUREMENT-BASED TRAFFIC MODELING 22
H. Che and S.-Q. Li

DYNAMIC CHANNEL ASSIGNMENT FOR WDM OPTICAL NETWORKS WITH LITTLE OR NO WAVELENGTH CONVERSION 32
O. Gerstel, G. Sasaki, and R. Ramaswami

ALGORITHMS FOR ESTABLISHING AND MAINTAINING SESSIONS IN WIRELESS NETWORKS WITH CHANGING CONNECTIVITY 44
A. Michail and A. Ephremides

PREDICTIVE QUEUEING MULTIPLE ACCESS– A WIRELESS ATM PROTOCOL FOR MULTIMEDIA COMMUNICATION 54
J.Y. Hui

I-B: SOURCE CODING
Chairman: K. Zeger
(University of California, San Diego)

EXISTENCE OF OPTIMAL PREFIX CODES FOR INFINITE SOURCE ALPHABETS 62
T. Linder, V. Tarokh, and K. Zeger

IDENTIFICATION PLUS TRANSMISSION OVER CHANNELS WITH PERFECT FEEDBACK 66
S. Venkatesan and V. Anantharam

EMBEDDED CODES FOR COMBINED SOURCE-CHANNEL CODING 74
S. Gadkari and K. Rose

THE COMMON RANDOMNESS CAPACITY OF A PAIR OF DISCRETE MEMORYLESS CHANNELS 84
S. Venkatesan and V. Anantharam

TRANSMISSION ENERGY ALLOCATION FOR ROBUST MULTI-STAGE VECTOR QUANTIZATION 92
S. Gadkari and K. Rose

AN AUTOMATIC REPEAT REQUEST SCHEME FOR JOINT LOSSLESS-SOURCE AND CHANNEL CODES 102
G.F. Elmasry

I-C: ROBUST AND OPTIMAL CONTROL I
Chairman: B. Bamieh
(University of Illinois at Urbana-Champaign)

STRONG STABILIZATION AND H^{∞} OPTIMIZATION FOR SISO SYSTEMS 112
M.J. Smith and A.E. Frazho

ROBUST STABILIZATION FOR SYSTEMS WITH RANK-ONE UNCERTAINTY STRUCTURE.....122
J. Lu and M.A. Rotea

ROBUST STABILITY OF UNCERTAIN FUNCTIONAL DIFFERENTIAL EQUATIONS 132
A.S.C. Sinha

ROBUST PERFORMANCE BOUNDS BASED ON LYAPUNOV FUNCTIONS FOR UNCERTAIN SYSTEMS 142
V. Balakrishnan

A NEW FORMULATION OF SOME DISCRETE-TIME STOCHASTIC-PARAMETER STATE ESTIMATION PROBLEMS.....152
E.E. Yaz and Y.I. Yaz

PROBABILISTIC ENHANCEMENT OF ROBUSTNESS MARGINS PROVIDED BY LINEAR MATRIX INEQUALITIES.....160
B.R. Barmish, C.M. Lagoa, and P.S. Shcherbakov

I-D: CDMA COMMUNICATION SYSTEMS
Chairman: R.E. Blahut
(University of Illinois at Urbana-Champaign)

FURTHER RESULTS FOR MULTI-RATE DECORRELATORS FOR SYNCHRONOUS DS/CDMA SYSTEMS.....170
J. Chen and U. Mitra

BLIND DECORRELATION OF ASYNCHRONOUS MULTIUSER SIGNALS USING FRACTIONALLY-SPACED SAMPLING.....180
M. Brandt-Pearce

MULTIUSER DECISION-FEEDBACK RECEIVERS FOR THE GENERAL GAUSSIAN MULTIPLE-ACCESS CHANNEL.....190
T. Guess and M.K. Varanasi

DESIGN AND PERFORMANCE OF CODED HIGH-DENSITY SYNCHRONOUS CDMA SYSTEMS.....200
J.C. McCanless, S.G. Wilson, and T.R. Giallorenzi

CAPACITY ANALYSIS OF CDMA WITH NON-EQUAL CELL LOADING AND SIZES.....210
S.W. Shin, S.M. Kwon, and P.S. Min

II-A: NETWORKS II
Organizer and Chairman: R. Srikant
(University of Illinois at Urbana-Champaign)

STABILITY OF POLLING SYSTEMS WITH STATE-INDEPENDENT ROUTING.....220
S. Foss, N. Chernova, and A. Kovalevskii

A GRAPH BASED APPROACH TO REDUCE COMPLEXITY IN QUEUING ANALYSIS.....228
G. Hjalmtysson

A GENERAL FRAMEWORK FOR ANALYZING SCHEDULERS AND REGULATORS IN INTEGRATED SERVICES NETWORKS.....239
R. Agrawal and R. Rajan

A DYNAMIC SCHEDULING COMMUNICATION PROTOCOL AND ITS ANALYSIS FOR HYPERCUBES.....249
A. Banerjee and E. Varvarigos

ASYMPTOTIC OPTIMALITY OF A MEASUREMENT-BASED CALL ADMISSION CONTROL SCHEME 259
D. Tse

A PRICING MODEL FOR NETWORKS WITH PRIORITIES 269
H. Jiang, I. Sidhu, and S. Jordan

II-B: CODING THEORY I

Organizers: A. Vardy and R.E. Blahut
(University of Illinois at Urbana-Champaign)
Chairman: R.E. Blahut
(University of Illinois at Urbana-Champaign)

ASYMPTOTICALLY GOOD SEQUENCES OF CURVES AND CODES 276
R. Pellikaan

THE WEIGHT HIERARCHIES AND CHAIN CONDITION OF A CLASS OF CODES FROM VARIETIES OVER FINITE FIELDS 286
X. Wu, G.-L. Feng, and T.R.N. Rao

CONSTRUCTION AND DECODING OF A NEW CLASS OF DOUBLE-ERROR-CORRECTING CODES FROM HERMITIAN CURVES 294
B.-Z. Shen and K.K. Tzeng

ON THE CLASSIFICATION OF SELF-DUAL CODES 302
W.C. Huffman

TECHNIQUES USED IN THE CLASSIFICATION OF SELF-DUAL Z_4 CODES 312
J. Fields and V. Pless

LATTICES AND CODES OVER RINGS 322
I.M. Duursma and J.J. Boutros

II-C: ROBUST AND OPTIMAL CONTROL II

Chairman: P. Voulgaris
(University of Illinois at Urbana-Champaign)

A NEW EFFICIENT METHOD FOR OPTIMIZATION OF A CLASS OF NONLINEAR DYNAMIC SYSTEMS WITHOUT LAGRANGE MULTIPLIERS 325
S.K. Agrawal and N. Faiz

GLOBALLY OPTIMAL FEEDBACK CONTROL OF NONLINEAR SYSTEMS: PART I, THE CONSTANT CONTROL MATRIX 333
S.-P. Shue, M.E. Sawan, and K. Rokhsaz

RELATIONSHIPS IN MODERN STOCHASTIC CONTROL: RISK-SENSITIVE AND MINIMAL COST VARIANCE CONTROL 343
C.-H. Won, M.K. Sain, and B.F. Spencer, Jr.

ROBUST CONTROL OF MARKOV DECISION PROCESSES AND CONNECTION TO RISK-SENSITIVE CONTROL 353
S. Coraluppi and S. Marcus

A SUFFICIENT CONDITION FOR THE STABILITY OF TIME-VARYING STATE-SPACE SYSTEMS: A GENERALIZATION OF THE CAYLEY-HAMILTON THEOREM 363
P.R. Bouthellier

A CONTROL LYAPUNOV FUNCTION APPROACH TO ROBUST STABILIZATION OF NONLINEAR SYSTEMS .. 372
M.W. McConley, B.D. Appleby, M.A. Dahleh, and E. Feron

II-D: LEARNING STRATEGIES AND COMPUTATIONAL METHODS FOR ADAPTIVE FILTERS
Organizer: W.K. Jenkins
(University of Illinois at Urbana-Champaign)
Chairman: D.L. Jones
(University of Illinois at Urbana-Champaign)

BLIND ERROR-BOUND ESTIMATION FOR OBE IDENTIFICATION 382
M. Nayeri, T.M. Lin, and J.R. Deller, Jr.

TRACKING OF TIME-VARYING PARAMETERS USING OPTIMAL BOUNDING ELLIPSOID ALGORITHMS 392
S. Kapoor, S. Gollamudi, S. Nagaraj, and Y.F. Huang

A SIMPLE WAY TO ADJUST THE LMS ALGORITHM STEP SIZE 402
Y. Dhiantravan and R. Priemer

IMPROVED DELTA LEVINSON AND SCHUR TYPE RLS ALGORITHMS 412
H. Fan

ADAPTIVE EQUALIZATION TECHNIQUES FOR MULTICARRIER CHANNELS 422
J.C. Strait

PLENARY LECTURE

THE VITERBI ALGORITHM-TWENTY-FIVE YEARS LATER:
THE FORWARD-BACKWARD ALGORITHM 432
G.D. Forney, Jr.

III-A: NETWORKS III
Chairman: V. Anantharam
(University of California, Berkeley)

A TWO-PHASE AUCTION ALGORITHM FOR HIGHLY ASYMMETRIC ASSIGNMENT PROBLEMS 447
N. Cohen and J. Brassil

FAIRNESS IN CELLULAR MOBILE NETWORKS 457
S. Sarkar and K.N. Sivarajan

STATISTICAL MULTIPLEXING OF ASYNCHRONOUS REGULATED TRAFFIC 469
H. Jin, R.J. McEliece, and Z. Yu

OPTIMAL DISTRIBUTED ROUTING IN NETWORKS WITH RANDOMLY-VARYING CONNECTIVITY 479
V. Tabatabaee and L. Tassiulas

III-B: CODING THEORY II
Organizers: A. Vardy and R.E. Blahut
(University of Illinois at Urbana-Champaign)
Chairman: A. Vardy
(University of Illinois at Urbana-Champaign)

PROBABILITY PROPAGATION AND ITERATIVE DECODING 482
B.J. Frey and F.R. Kschischang

SOME HIGH-RATE NEAR CAPACITY CODECS FOR THE GAUSSIAN CHANNEL 494
J.-F. Cheng and R.J. McEliece

CAN SEQUENTIAL DECODING COMPETE WITH TURBO-CODES? 504
D.J. Costello, Jr. and H.A. Cabral

III-C: CURRENT TOPICS IN ADAPTIVE SIGNAL PROCESSING
Organizer: W.K. Jenkins
(University of Illinois at Urbana-Champaign)
Chairman: G.A. Williamson
(Illinois Institute of Technology)

OPTIMAL ADAPTIVE FILTERING USING CROSS-VALIDATION 506
R.D. Nowak

A MARKOV ANALYSIS OF THE SECOND-ORDER DIGITAL PHASE LOCKED LOOP 516
M.P. Fitz, R. Balasubramanian, and J.V. Krogmeier

HOW GOOD IS THE ADAPTIVE POLYPHASE STRUCTURE? 526
M. Nayeri and U. Iyer

CONSTRAINED ADAPTATION OF SUBBAND ADAPTIVE FILTERS TO MINIMIZE ALIASING EFFECTS 536
G.A. Williamson and S. Zimmermann

JOINT IMPULSE RESPONSE SHORTENING 546
P.J.W. Melsa, R.C. Younce, and C.E. Rohrs

III-D: AEROSPACE APPLICATIONS
Organizers: B. Bamieh and S.P. Meyn
(University of Illinois at Urbana-Champaign)
Chairman: S.P. Meyn
(University of Illinois at Urbana-Champaign)

NONLINEAR, HIGH-DIMENSIONAL ADAPTIVE SIGNAL PROCESSING WITH AN APPLICATION TO ACTIVE NOISE CONTROL 556
R.H. Burkhart

III-E: MULTIDIMENSIONAL DIGITAL SIGNAL PROCESSING
Chairman: D.C. Munson, Jr.
(University of Illinois at Urbana-Champaign)

GRAY CONNECTED COMPONENTS FOR STEREO CORRESPONDENCE PROBLEM 566
Y. Wang and P. Bhattacharya

DELTA-OPERATOR LINEAR PREDICTION MODELS FOR TWO-DIMENSIONAL DATA 574
L.-H. Yiin and H.V. Poor

CONNECTIONS BETWEEN MULTIDIMENSIONAL SYSTEMS THEORY AND ALGEBRAIC GEOMETRY 583
C. Peterson, J. Rosenthal, and P.A. Weiner

CALIBRATION-FREE VISION BASED CONTROL OF A ROBOTIC MANIPULATOR 593
B.K. Ghosh, D. Xiao, N. Xi, and T.J. Tarn

IV-A: WIRELESS COMMUNICATIONS I
Organizer and Chairman: U. Madhow
(University of Illinois at Urbana-Champaign)

ADAPTIVE MULTIUSER DETECTION IN FADING CHANNELS 603
H.V. Poor and X. Wang

RECURSIVE MULTIUSER EQUALIZATION FOR CDMA SYSTEMS IN FADING ENVIRONMENTS 613
S.H. Isabelle and G.W. Wornell

DESIGN AND IMPLEMENTATION ISSUES FOR A WIDEBAND, INDOOR, DS-CDMA SYSTEM PROVIDING MULTIMEDIA ACCESS 623
C. Teuscher, S. Sheng, I. O'Donnell, K. Stone, and R. Brodersen

DECODING METRICS FOR THE IS-95 REVERSE LINK 633
C.D. Frank and F. Ling

PERFORMANCE EVALUATION OF A BLIND ADAPTIVE LINEAR RECEIVER FOR CELLULAR DS/CDMA SYSTEMS 642
T.F. Wong, T.M. Lok, and J.S. Lehnert

BIT ALLOCATION ALGORITHMS FOR FH-CDMA WIRELESS COMMUNICATION SYSTEMS 652
C.C. Wang and G.J. Pottie

IV-B: CODING THEORY III
Organizers: A. Vardy and R.E. Blahut
(University of Illinois at Urbana-Champaign)
Chairman: A. Vardy
(University of Illinois at Urbana-Champaign)

QUANTUM ERROR CORRECTION VIA CODES OVER GF(4) 662
A.R. Calderbank, E.M. Rains, P.W. Shor, and N.J.A. Sloane

PROBABILISTIC CRISSCROSS ERROR CORRECTION 673
R.M. Roth

COVERING LISTS IN MAXIMUM LIKELIHOOD DECODING 683
I. Dumer

COMPACT DESCRIPTION OF GOLAY SEQUENCES AND THEIR EXTENSIONS 693
R. Urbanke and A.S. Krishnakumar

CLASSES OF COSTAS AND SONAR SEQUENCES FOR MULTIPLE-TARGET RECOGNITION 703
O. Moreno and S. Maric

DENSE MORDELL-WEIL LATTICES OF ELLIPTIC CURVES OVER FUNCTION FIELDS: EXPLICIT COMPUTATION 707
V. Tarokh and I.F. Blake

IV-C: LARGE DEVIATIONS THEORY AND ITS APPLICATIONS
Organizer: S.P. Meyn
(University of Illinois at Urbana-Champaign)
Organizer and Chairman: J.S. Sadowsky
(Arizona State University)

BUFFER OVERFLOW PROBABILITIES IN MULTICLASS ATM SWITCHES 712
D. Bertsimas, I.C. Paschalidis, and J.N. Tsitsiklis

FUNCTIONAL LARGE DEVIATION PRINCIPLES FOR QUEUES 722
A.A. Puhalskii and W. Whitt

SHARING BANDWIDTH IN ATM 732
K. Ramanan and A. Weiss

ON THE RELEVANCE OF LONG-TAILED DURATIONS FOR THE STATISTICAL MULTIPLEXING OF LARGE AGGREGATIONS 741
N.G. Duffield

ON THE TYPICAL PATH TO HITTING A RARE SET 751
T. Zajic

NUMERICAL COMPUTATION OF LARGE DEVIATIONS EXPONENTS VIA SIMULATION 760
P.W. Glynn

IV-D: DISCRETE EVENT SYSTEMS
Organizer and Chairman: R.S. Sreenivas
(University of Illinois at Urbana-Champaign)

EXTENSIONS OF GSMP MODELS FOR BROADBAND NETWORK TRAFFIC MODELING 768
C.A. Brooks

SYNTHESIZING CONTROL CODE FOR ACHIEVING STATE TARGETS IN COMPOSED SYSTEMS 776
L.E. Holloway and S. Rajaram

A BISIMULATION APPROACH TO THE SUPERVISORY CONTROL OF DISCRETE EVENT SYSTEMS 786
G. Barrett and S. Lafortune

CONTROL OF TIMED DISCRETE-EVENT SYSTEMS – A FRAMEWORK 796
T.-J. Ho

A SUPERVISOR SYNTHESIS ALGORITHM FOR STOCHASTIC DISCRETE EVENT SYSTEMS 806
R.C.H. Chang and R.H. Kwong

RULE-BASED MODELING OF HYBRID DYNAMIC SYSTEMS USING PETRI NETS 815
M. Gotesman and N. Lopez Benitez

V-A: WIRELESS COMMUNICATIONS II
Organizer: U. Madhow
(University of Illinois at Urbana-Champaign)
Chairman: L. Tassiulas
(University of Maryland, College Park)

ERROR CONTROL AND ENERGY CONSUMPTION IN WIRELESS COMMUNICATIONS CHANNELS 825
M. Zorzi and R.R. Rao

TIME DEPENDENT OUTAGES IN CELLULAR SYSTEMS: MINIMUM DURATION OUTAGES 835
N.B. Mandayam, P.-C. Chen, and J.M. Holtzman

POWER CONTROL USING STOCHASTIC MEASUREMENTS 845
S. Ulukus and R.D. Yates

LOCATION TRACKING MECHANISMS FOR OPTIMAL MOBILITY ADAPTATION 855
L. Tassiulas and F.M. Anjum

PERFORMANCE ANALYSIS OF CELLULAR HYBRID CHANNEL ALLOCATION USING THE INTERRUPTED POISSON PROCESS 865
J.M. Harris and S.P. Kumar

V-B: CODING THEORY IV
Chairman: G.D. Forney, Jr.
(Motorola, Inc.)

THE MULTICOVERING RADII OF CODES EXTENDED ABSTRACT 875
A. Klapper

CONVOLUTIONAL CODES, R-CYCLIC CODES, AND LINEAR SYSTEMS 885
E.V. York

ANALYSIS OF CONVOLUTIONAL ENCODERS VIA GENERALIZED SYLVESTER MATRICES AND STATE SPACE REALIZATIONS 893
B.M. Allen and J. Rosenthal

V-C: NETWORKS IV
Chairman: B. Hajek
(University of Illinois at Urbana-Champaign)

LONG-TERM DEPENDENT VBR VIDEO MODEL USING ON-OFF SOURCES 903
Q. Xiao and E.J. Coyle

FRACTIONAL BROWNIAN MOTIONS AND LEVY MOTIONS AS LIMITS OF STOCHASTIC TRAFFIC MODELS 913
T. Konstantopoulos and S.-J. Lin

ASYMPTOTICS FOR M/G/1 LOW-PRIORITY WAITING-TIME TAIL PROBABILITIES 923
J. Abate and W. Whitt

V-D: FUZZY CONTROL
Chairman: M. Jamshidi
(University of New Mexico)

INTRODUCTION TO FUZZY SYSTEMS: FROM FUZZY SETS TO INDUSTRIAL APPLICATIONS 933
M. Jamshidi

CONCURRENT FL-PI CONTROLLER DESIGNED USING GAs 942
G. Li, K.M. Tsang, and H. Li

APPLICATION OF PEAK PATTERN TECHNIQUE TO DESIGN FUZZY CONTROLLERS TO MEET CLOSED-LOOP FUZZY SPECIFICATIONS 950
C. Kolodziej and R. Priemer

V-E: PROCESS CONTROL
Chairman: R. Braatz
(University of Illinois at Urbana-Champaign)

A SOLUTION TO GENERALIZED CONSTANT VALUE CONTROL PROBLEMS OF PLANTS WITH DOMINANT TIME DELAY 960
G. Li, K.M. Tsang, and S.L. Ho

AN INTEGRATED IDENTIFICATION AND CONTROL PROCEDURE FOR SHEET AND FILM PROCESSES 970
A.P. Featherstone and R.D. Braatz

PERFORMANCE ASSESSMENT OF INTERACTING PROCESSES 980
O. Taha, G.A. Dumont, and J.B. Ball

V-F: ADAPTIVE CONTROL AND NEURAL NETWORKS
Chairman: P.R. Kumar
(University of Illinois at Urbana-Champaign)

RISK-SENSITIVE IDENTIFICATION OF DYNAMIC SYSTEMS BY NEURAL NETWORKS 982
J.T.-H. Lo

RISK-SENSITIVE APPROXIMATION OF FUNCTIONS BY NEURAL NETWORKS 991
J.T.-H. Lo

NOISY RECURRENT NEURAL NETWORKS: THEORY AND QUANTITATIVE PERFORMANCE EVALUATION 997
O. Olurotimi and S. Das

DISTRIBUTED ESTIMATION OF LOCATION PARAMETER OF TWO EXAMPLE DENSITIES 1009
M. Su, J. Cheng, and R. Viswanathan

INDEX OF AUTHORS 1019

A Control Policy for the Stabilization of End-to-End Congestion: Multiple Sources Case

Hongyi Zhang and Oliver W. Yang
Department of Electrical Engineering, University of Ottawa
Ottawa, Ontario, Canada K1N 6N5
H. Mouftah
Department of Electrical Engineering, Queen's University
Kingston, Ontario, Canada K7L 3N6

Abstract

In this paper, a discrete time fluid model is used to model a high-speed switching network. The most important features in our model are that 1) both the best effort traffic and guaranteed traffic can be intergrated into the switch, and 2) the number of active virtual connections can be changed over time. Based on this model, a robust congestion controller is proposed for the flow regulation of the best-effort (ABR) traffic in high-speed networks. The controller only requires the knowledge of the buffer occupancy, and is not dependent on the change of the number VCs during the process. It is shown that the system can be piecewisely asymptotically stabilized without oscillation.

1 Introduction

High-speed ATM networks are expected to support multimedia traffic consisting of a variety of traffic classes, each with a different quality of service requirement. Two basic classes of service are under consideration for ATM networks: reserved traffic with guaranteed service, and best-effort traffic with no explicit guaranteed service. Correspondingly, two classes of congestion control strategies have been discussed for high-speed ATM networks: open-loop control, and closed-loop control.

Among all proposed congestion control mechanisms for the best-effort traffic, the closed-feedback rate-based control that allows the network to control the source rate at each source node offers the most effective solution to bandwidth sharing among all competing users. Typical examples of the rate-based control schemes are forward explicit congestion notification (FECN) and backward explicit congestion notification (BECN) [9]. These control strategies are well-known and very effective in conventional packet-switched networks. However, since the propagation delay is long compared to the transmission time of a data unit in high-speed networks, the controller for each virtual connection may receive out-of-date information on the network state. Thus the simple proportional scheme may cause the whole network to operate at an unstable point. This yields the notorious oscillation problem that greatly degrades the network performance.

In this paper, we extend our results in [11] to consider a controller that is geographically distributed over the source nodes. Based on the feedback information on the

buffer occupancy in the bottleneck switch, the local controller can calculate the maximum rate at which the source can transmit data to the network. It is shown that our control scheme not only stabilizes the network, but also has the properties: 1) it serves both guaranteed traffic and best-effort traffic, 2) it is fair for each virtual connection, which means that each source always tries to catch up its own allocated transmission rate without interference from other source nodes, and 3) it has high utilization, which means the source can always transmit at the maximum allowed rate.

2 Network Modeling and Assumptions

We now consider a single switching node which accumulates multiple virtual connections (VC). The number of controlled VCs which go through the switching node are equal to the number of active source nodes to be controlled. This number may vary with time t. The maximum number of virtual connections which are allowed at the node is N plus one which represents a group of guaranteed service traffic streams $w(t)$. The switching node has a finite buffer space K to store the incoming cells, and its output link has a constant data rate ν.

There are two kinds of delays for VC's: $\tau_{1i}^{'}$ is the path delay from the ith source node to the switching node and $\tau_{2i}^{'}$ is the path delay from the switching node to the ith source node. There are two kinds of traffic: 1) uncontrolled traffic (e.g., guaranteed service traffic), which is not throttled at the source node as long as it conforms to specifications, and 2) controlled traffic (e.g., best-effort service traffic), which can only be transmitted when there is no congestion in the network. Let $w(t)$ be the total uncontrolled traffic rate, and let $q_i(t)$ be the ith controlled traffic rate. $q_i(t)$ is regulated based on buffer occupancy $x(t)$ which is sent to the controlled source nodes every T seconds.

According to the modeling assumptions, the dynamics of a switching node in a network can be described by a non-linear, time-variant, and differential equation with path delays, i.e.,

$$\dot{x}(t) = Sat_K\{\sum_{i=1}^{N} e_i(t-\tau_{1i}^{'})q_i(t-\tau_{1i}^{'}) + w(t) - \nu\} \tag{1}$$

where K is the buffer size, $x(t)$ is the buffer occupancy, and

$$Sat_K\{x\} = \begin{cases} K & x > K \\ x & 0 \le x \le K \\ 0 & x < 0 \end{cases}$$

$$e_i(t) = \begin{cases} 1 & \text{the } i\text{th source node is active} \\ 0 & \text{otherwise.} \end{cases}$$

In the system described by (1), the ith active source node can only measure the buffer occupancy $x(t)$ after a path delay $\tau_{2i}^{'}$. It implies that if the feedback control is

employed, the system has to include input functions involving the round trip delays $\tau_i^{'} := \tau_{1i}^{'} + \tau_{2i}^{'}$, $i = 1, 2, \cdots, N$. Let $u_i(t) = q_i(t + \tau_{2i}^{'})$, then equation (1) can be rewritten as

$$\dot{x}(t) = Sat_K\{\sum_{i=1}^{N} e_i(t - \tau_{1i}^{'})u_i(t - \tau_i^{'}) + w(t) - \nu\} \ . \tag{2}$$

Now assume that the switching node sends information to the source nodes every T seconds, and that path delay factors $\tau_{1i}^{'}/T$ can be decomposed as $\tau_{1i} + \gamma_i/T$, where τ_{1i} is an integral multiple of T and is called the feedforward delay from the source node to the bottleneck node, and $0 \leq \gamma_i < T$, $i = 1, 2, \cdots, N$. Then the change of buffer occupancy of the switching node in Fig. 1 is given by the following discrete-time expression

$$x(n+1) = Sat_K\{x(n) + \sum_{i=1}^{N} e_i(n - \tau_{1i} + \frac{\gamma_i}{T})\lambda_i(n - \tau_{1i} + \frac{\gamma_i}{T}) + d(n) - \mu\} \tag{3}$$

where $\lambda_i(n) = Tq(nT)$ and $d(n) = Tw(nT)$ denote respectively the numbers of cells flowing into the network from the ith VC and flowing into the switch from the uncontrolled traffic during the the nth interval of T, and $\mu = T\nu$ denotes the number of cells emitted from the switching node during the the nth interval of T.

In the following, we consider the case where the path delay $\tau_{1i}^{'}$ is a multiple of T, i.e., $\gamma_i = 0$, $i = 1, 2, \cdots, N$. This is reasonable because we can always add a small delay to the path delay by holding the cells at source nodes. The buffer equation given in (3) can then be rewritten as

$$x(n+1) = Sat_K\{x(n) + \sum_{i=1}^{N} e_i(n - \tau_{1i})\lambda_i(n - \tau_{1i}) + d(n) - \mu\} \tag{4}$$

We similarly assume that the path delay $\tau_{2i}^{'}$ is a multiple of T, i.e., $\tau_{2i}^{'} = \tau_{2i}T$, $i = 1, 2, \cdots, N$. The integer τ_{2i} is then called the feedback delay from the bottleneck node to the source node. Define

$$u_i(n) = \lambda_i(n + \tau_{2i}) - \mu_i$$

where μ_i is the maximum number of cells the ith source node is allowed to transmit the data into the network in an interval T.

Let $\hat{\mu} = \mu - \sum_{i=1}^{N} e_i(n - \tau_{1i})\mu_i$. We then have from (3),

$$x(n+1) = Sat_K\{x(n) + \sum_{i=1}^{N} e_i(n - \tau_{1i})u_i(n - \tau_i) + d(n) - \hat{\mu}\} \tag{5}$$

where $\tau_i := \tau_{1i} + \tau_{2i}$ is called the total delay for the ith VC.

In comparison with conventional fluid models for a switch network described in [1]$\sim$ [7], the most important features added to our model are that 1) both best-effort service traffic and guaranteed service traffic are integrated into the switch, and 2) the number of active VCs can be changed over time as the factors e_i are time-variant.

3 The Robust Control Algorithm

A robust congestion controller is proposed here in the sense that it only requires the information on buffer occupancy without knowing the change of the number of VCs during the network operation. Assume that μ_i/T is the maximum rate allocated for the ith VC with $\sum_{i=1}^{N} \mu_i \leq \mu$. Let $t_i = n_i T$, $i = 1, 2, \cdots$, be the points where the number of VCs is changed. Therefore the number of VCs is constant over the interval $\mathcal{T}_i := [t_i,\ t_{i+1})$. Also let the set $\mathcal{N}_{\mathcal{T}_i} := \{k,$ the kth VC is active over $\mathcal{T}_i\ \}$. We will show that the following algorithm can achieve stability for the network described by (5).

The Control Algorithm:

For $i \in \mathcal{N}_{\mathcal{T}_i}$,

$$\begin{aligned} u_i(n) &= -Sat_{\mu_i}\{\frac{1}{N}x(n-\hat{\tau}_i) + \sum_{j=1}^{\tau} u_i(n-j)\} \\ \lambda_i(n) &= \mu_i + u_i(n-\tau_{2i}) \end{aligned} \tag{6}$$

where $\tau = max\{\tau_1, \cdots, \tau_N\}$, and $\hat{\tau}_i$ is chosen such that $\hat{\tau}_i + \tau_i = \tau$ for $i \in \mathcal{N}_{\mathcal{T}_i}$.

4 Stability Analysis

Definition 1 *A switching network described by (5) is said to have a slow temporal change in load if the frequency and the change in the number of VCs are small.*

In the following, we would only consider the locally asymptotical stability properties of the network (5) with a slow temporal change of load along with the congestion controller (6). The z-transform method is used, where the system's initial conditions are assumed to be zero because the initial conditions will not affect the system's stability.

For any interval $\mathcal{T}_l$, $l = 1, 2, \cdots$, we consider a node with multiple controlled inputs described by:

$$x(n+1) = x(n) + \sum_{i=1}^{N} e_i(n-\tau_{1i})u_i(n-\tau_i) + d(n) - \hat{\mu} \tag{7}$$

In general, the traffic controller at each source node can be represented by

$$u_i(n) = \sum_{j=1}^{\tau} b_{\tau-j}u_i(n-j) + \sum_{j=0}^{\hat{\tau}_i} a_{i(\hat{\tau}_i-j)}x(n-j) \tag{8}$$

where b_j and a_{ij} are coefficients, $\tau := \max\{\tau_1, \tau_2, \cdots, \tau_N\}$, and $\hat{\tau}_i$ $(< \tau)$ is a positive integer to be determined.

Since the number of currently-active VCs is fixed during the interval $\mathcal{T}_l$, the z transform of (7) is

$$(z-1)X(z) = \sum_{i=1}^{N} e_i z^{-\tau_i} U_i(z) + D(z) - \hat{\mu}R(z) \tag{9}$$

where $X(z) := \sum_{n=n_l}^{\infty} x(n) z^{n-n_l}$, and $U_i(z)$, $D(z)$ and $R(z)$ are similarly defined as in $X(z)$.

Let

$$W_\tau(z) := \begin{bmatrix} z^\tau \\ \vdots \\ z \\ 1 \end{bmatrix}, \quad B_\tau := \begin{bmatrix} b_\tau \\ \vdots \\ b_1 \\ b_0 \end{bmatrix}, \quad A_{\hat{\tau}_i} := \begin{bmatrix} a_{i\hat{\tau}_i} \\ \vdots \\ a_{i1} \\ a_{i0} \end{bmatrix}$$

Then the z transform of (8) is given by

$$U_i(z) = -z^{\tau-\hat{\tau}_i}(B_\tau^T W_\tau(z))^{-1} A_{\hat{\tau}_i}^T W_{\hat{\tau}_i}(z) X(z) \tag{10}$$

where b_τ in B_τ is -1. Substituting (10) into (9), we obtain

$$\Delta(z)X(z) = B_\tau^T W_\tau(z)[D(z) - \hat{\mu} R(z)] \tag{11}$$

where $\hat{\mu} = \mu - \sum_{i=1}^{N} e_i \mu_i$, and $\Delta(z)$ is the characteristic equation of the system given by

$$\Delta(z) = (z-1)B_\tau^T W_\tau(z) + \sum_{i=1}^{N} z^{\tau-\hat{\tau}_i-\tau_i} e_i A_{\hat{\tau}_i}^T W_{\hat{\tau}_i}(z)$$

Choose $\hat{\tau}_i + \tau_i = \tau$, $i \in \mathcal{N}_{\mathcal{T}_l}$ so that $\Delta(z)$ becomes a polynomial function of z. We obtain

$$\Delta(z) = (z-1)B_\tau^T W_\tau(z) + \sum_{i=1}^{N} e_i A_{\hat{\tau}_i}^T W_{\hat{\tau}_i}(z) \tag{12}$$

The roots of $\Delta(z) = 0$ are determined by the coefficients of b_j, $j = 0, 1, \cdots, \tau - 1$, and a_{ij}, $j = 1, 2, \cdots, \hat{\tau}_i$, $i \in \mathcal{N}_{\mathcal{T}_l}$.

Theorem 1 *Assume that each source node only obtains feedback information on the buffer occupancy of the congestion node. Then, the network described by (7) with a slow temporal change of load, can be piecewisely stabilized by the controller (8) if*

$$\begin{array}{ll} a_{i0} = -\dfrac{1}{N} & i \in \mathcal{N}_{\mathcal{T}_l}, \quad l = 1, 2, \cdots \\ a_{ij} = 0 & j = 1, 2, \cdots, \hat{\tau}_i, \quad i \in \mathcal{N}_{\mathcal{T}_l}, \quad l = 1, 2, \cdots \\ b_j = 0, & j = 0, 1, \cdots, \tau - 1 \end{array} \tag{13}$$

Proof. Omitted.

5 System Steady Behavior

Since it has been shown that the switch network can be piecewisely stabilized by controller (6), the system steady states can be achieved under assumptions that the uncontrolled traffic rate $d(n)$ and the number of VCs tend to be constant, as $n \to \infty$.

Let x_s, u_{is}, λ_{is} and d_s be the steady state values corresponding to $x(n)$, $u_i(n)$, $\lambda_i(n)$ and $d(n)$ respectively in equations (5) and (6). Then as $n \to \infty$,

$$\begin{aligned} x_s &= Sat_K\{x_s + \sum_{i=1}^{N} e_i u_{is} + d_s - \hat{\mu}\} \\ u_{is} &= -Sat_{\mu_i}\{\frac{1}{N} x_s + \tau u_{is}\} \\ \lambda_{is} &= \mu_i + u_{is},\ i \in \mathcal{N}_s \end{aligned} \tag{14}$$

where $\mathcal{N}_s$ is the set containing all active VCs as $n \to \infty$.

The following result gives the solution to the above equations.

Theorem 2 *The steady behavior of the system described by (5) and (6) is given by*
a) if $d_s < \hat{\mu}$,

$$x_s = 0,\ \ u_{is} = 0,\ \textit{and}\ \lambda_{is} = \mu_i,\ i \in \mathcal{N}_s$$

b) if $\hat{\mu} \le d_s \le \hat{\mu} + min\{M_s\mu_0,\ \frac{KM_s}{N(\tau+1)}\}$,

$$\begin{aligned} x_s &= \frac{N(\tau+1)}{M_s}(d_s - \hat{\mu}) \\ u_s &:= u_{is} = -\frac{1}{M_s}(d_s - \hat{\mu}) \\ \lambda_{is} &= \mu_i + u_s,\ \ i \in \mathcal{N}_s \end{aligned} \tag{15}$$

where M_s *is the steady state value of* M *as* $n \to \infty$.

Proof. Omitted.

Remark 1 *It is observed from Theorem 2 that the traffic rate at each source node is regulated only based on the uncontrolled traffic* d_s, *and that all* u_{is} *are equal to* u_s. *This shows that the source nodes are treated fairly, and they can share the switch's bandwidth without interference to each other. In order to design a lossless data network, it is required that the buffer size* K *is at least greater than* $\frac{N(\tau+1)}{M_s}(d_s - \hat{\mu})$. *Clearly, the buffer occupance* x_s *in (15) will become large if* N *is large or* M_s *is small.*

6 Simulations

Since our congestion controller regulates the source rate over time, we are particularly interested in analyzing the transient and steady behaviors of the network. Transient behaviors such as the duration of response time and the maximum transient buffer occupancy are the main concern in performance analysis.

Simulation is performed on the model described by (5), where the bottleneck switch has a capacity $\nu = 2[illegible]0Mbps$. We choose the sampling time $T = 1msec$. The maximum

number of VCs which are allowed to enter the switch is $N = 5$. The forward path delays are $\tau_{11}^{'} = 1msec$, $\tau_{12}^{'} = 2msec$, $\tau_{13}^{'} = 2msec$, $\tau_{14}^{'} = 3msec$, and $\tau_{15}^{'} = 3msec$ respectively. The feedback path delays are $\tau_{21}^{'} = 1msec$, $\tau_{22}^{'} = 2msec$, $\tau_{23}^{'} = 3msec$, $\tau_{24}^{'} = 4msec$, and $\tau_{25}^{'} = 4msec$ respectively. Thus, the maximum round-trip delay $\tau = 7$. In the following, Example 1 shows the case of all sources active while Example 2 shows the case of two active sources.

Example 1. Figure 1 shows transient and steady behaviors of the network fed by five active controlled sources with the maximum allocated rates $\mu_i/T = 40Mbps$, $i = 1, 2, 3, 4, 5$. In this case, all source nodes are active, and $\hat{\mu}/T = 20Mbps$.

Figure 1a shows the uncontrolled traffic rate $d(n)$ with $d_s/T = 20Mbps$. Figure 1b shows the controlled traffic rate $\lambda_i(n)/T$, $i = 1, 2, 3, 4, 5$. Figure 1c shows that the controlled source catch up with the maximum allowed rate such that the total input rate $[\sum_{i=1}^{5} \lambda_i(n) + d(n)]/T$ is equal to the switch capacity μ/T except that there are some small pulses. These pulses are produced due to the delay in the response of the controlled input $\lambda_i(n)$ to the uncontrolled input $d(n)$. The narrow pulses indicate that this control algorithm has a short response time. One would observe from Figure 1d that the steady state of buffer occupancy x_s is zero since $d_s = \hat{\mu}$. This verifies the results in Theorem 2.

Example 2. Figure 2 shows the transient and steady behaviors of the network fed by two active controlled sources. Assume that the active source nodes are source 1 and source 3, with the maximum allocated rates $\mu_1/T = 80Mbps$ and $\mu_3/T = 120Mpbs$ respectively. We have that $\hat{\mu} = 20$. It is observed from Figures 2b, 2c and 2d that the system response time is longer than that in Example 1. The maximum buffer occupancy is $800Kb$, which is equivalent to the steady state of buffer occupancy calculated from (15) in Theorem 2,

$$x_s = \frac{N(\tau + 1)}{M_s}(d_s - \hat{\mu}) = \frac{5 \times 8}{2} \times 40 = 800Kb$$

It is demonstrated from Figure 1c in Example 1 and Figure 2c in Example 2 that all controlled sources nodes always try to catch up their own possible bandwidth without interference from each other. This implies that our controller satisfies the fairness requirement. Also the total input rate displayed in Figure 1c and Figure 2c are close to the switch capacity. This implies that our controller has high utilization.

7 Conclusions

In this paper, a discrete-time fluid model was used to model a network which can accumulate both guaranteed traffic and best-effort traffic. The congestion controller is geographically distributed over the source nodes. Each local controller can determine the maximum allowed rate at which the source can transmit data to the network.

We have analyzed a robust congestion controller in the sense that the controller is insensitive to the change of number of VCs during the network operation. With this controller applied, we demonstrate that the network is piecewisely asymptotically stable and exhibits good transient behavior.

Acknowledgement:

This paper is financially supported by a TRIO Research Assistantship and NSERC Operating Grant under contract # OGP0042878.

References

[1] S. Keshav, "A control-theoretic approach to flow control," Proc. ACM SIGCOMM'91, pp. 3-15, 1991.

[2] K. Fendick, M. Rodrigues, and A. Weiss, "Analysis of a rate-based control strategy with delayed feedback," Proc. ACM SIGCOMM'92, pp. 136-148, 1992.

[3] H. Ohsaki et al., "Rate-based congestion control for ATM Networks," ACM Comp. Commun. Rev., vol. 35, pp.60-72, 1995.

[4] K.Y. Siu and H.Y. Tzeng, "Intelligent congestion control for ABR service in ATM networks," ACM Comp. Commun. Rev., vol. 35, pp. 81-106, 1995.

[5] D. Cavendish et al., "Proportional rate-based congestion control under long propagation delay," Int. J. Communication Systems, vol. 8, pp. 79-89, 1995. IEEE J. Select. Areas Comm., vol.13, pp. 1267-1283, 1995.

[6] L. Benmohamed and S.M. Meekov, "Feedback control of congestion in packet switching networks: The case of a single congested node," IEEE/ACM Trans. Networking, vol. 1, pp. 693-707, 1993

[7] M. Hluchyj and N. Yin, "On closed-loop rate control for ATM networks," Proc. IEEE INFOCOM, pp. 99-108, 1994.

[8] K. Ogata, "Discrete time control system," Prentica Hall, 1987.

[9] C. Q. Yang and A.A.S. Reddy, "A taxonomy for congestion control algorithms in packet switching networks," IEEE Network, pp. 34-45, 1995.

[10] F. Bonomi and K. Fendick, "The rate-based flow framework for the available bit rate ATM service," IEEE Network Mag., vol. 9, pp. 25-39, 1995.

[11] H. Zhang, O. Yang and H. Mouftah, "A controller for the stabilization of end-to-end congestion: single source case," Proc. of the 4th International Conference on Telecommunication Systems, pp. 340-346, 1996.

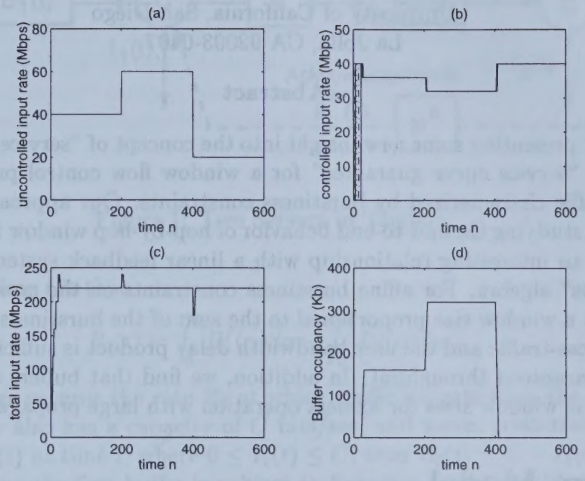


Figure 1: Five active controlled source nodes with $\mu_i = 40Mbps$, $i = 1, 2, 3, 4, 5$

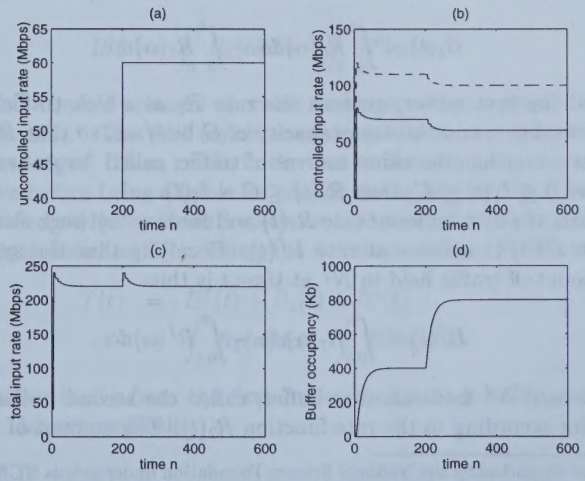


Figure 2: Two active controlled source nodes with $\mu_1 = 80Mbps$ and $\mu_3 = 120Mbps$

Service Guarantees for Window Flow Control[1]

R. L. Cruz C. M. Okino
Department of Electrical & Computer Engineering
University of California, San Diego
La Jolla, CA 92093-0407

Abstract

After presenting some new insight into the concept of "service curves," we derive a "service curve guarantee" for a window flow control protocol with cross-traffic characterized by burstiness constraints. Our approach is convenient for studying the end-to-end behavior of hop-by-hop window flow control, and has an interesting relationship with a linear feedback system under the "min-plus" algebra. For affine burstiness constraints on the cross-traffic, we find that a window size proportional to the sum of the burstiness parameters of the cross-traffic and the user bandwidth delay product is sufficient to maximize guaranteed throughput. In addition, we find that buffers need not be as large as window sizes for lossless operation with large propagation delays.

1 System Model

We will consider the two server system depicted in Figure 1, which models a window flow protocol. Traffic from a source is generated according to a function of time R_0, called a rate function, such that $R_0(t)$ is the instantaneous rate at which traffic is being generated at time t. The traffic from the source feeds a buffer, called the first buffer. Traffic departs the first buffer according to the rate function R_1. We assume that the system is empty at time 0. Let $B_1(t)$ denote the amount of traffic held in the first buffer at time t. Thus

$$B_1(t) = \int_0^t R_0(\alpha)d\alpha - \int_0^t R_1(\alpha)d\alpha \,. \tag{1}$$

A server, called the first server, governs the rate R_1 at which traffic departs the first buffer. The server has a transmission capacity of C bits/sec, so that $R_1(t) \leq C$ for all t. In fact, the first server handles other sources of traffic, called "cross-traffic," at time t at rate $I_1(t)$, where $0 \leq I_1(t) \leq C$; thus $R_1(t) \leq C - I_1(t)$.

Traffic departs the first buffer at rate $R_1(t)$ and feeds a "network element", N^f, which serves traffic in a FIFO manner at rate $R^f(t)$. Recalling that the system is empty at time 0, the amount of traffic held in N^f at time t is thus

$$B^f(t) = \int_0^t R_1(\alpha)d\alpha - \int_0^t R^f(\alpha)d\alpha \,. \tag{2}$$

Network element N^f feeds another buffer, called the second buffer. Traffic departs the second buffer according to the rate function $R_2(t)$. The amount of traffic held in the

[1]This work was supported by the National Science Foundation under grants NCR-91-58618 and NCR 94-15684, by the Army Research Office under grant FRI DAAH04-95-1-0248, and by the Air Force Office of Scientific Research under grant F49620-95-1-0538

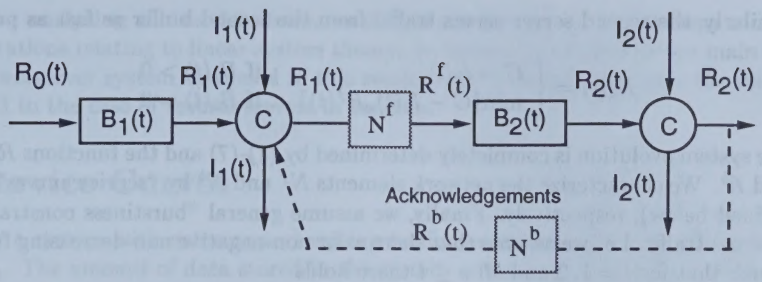


Figure 1: Two Servers in Tandem.

second buffer at time t is thus

$$B_2(t) = \int_0^t R^f(\alpha)d\alpha - \int_0^t R_2(\alpha)d\alpha \ . \tag{3}$$

A second server governs the rate R_2 at which traffic departs from the second buffer. The second server also has a capacity of C bits/sec, and serves cross-traffic from other sources at rate $I_2(t)$ at time t, where $0 \leq I_2(t) \leq C$; thus $R_2(t) \leq C - I_2(t)$.

Traffic departing the first buffer is subject to "window flow control," whereby traffic departing the first buffer may have to wait for acknowledgements from the the second server. More specifically, as traffic from the original source departs the second buffer, acknowledgements are correspondingly generated by the second server and sent back to the first server via a network element N^b. The rate at which acknowledgements are generated at the second server at time t is $R_2(t)$. The network element N^b operates in a FIFO manner and serves acknowledgements at rate $R^b(t)$. Thus, the amount of acknowledgements in N^b at time t is

$$B^b(t) = \int_0^t R_2(\alpha)d\alpha - \int_0^t R^b(\alpha)d\alpha \ . \tag{4}$$

The window flow control protocol operates with respect to a positive parameter K, called the "window size." The first server must insure that no more than K units of traffic are unacknowledged. Thus, traffic in the first buffer may have to wait for acknowledgements to arrive before being eligible for service at the first server. The total amount of traffic unacknowledged (sometimes known as the number of outstanding credits or tokens) at time t is denoted as $T(t)$. Using the definitions above, it follows that

$$\begin{aligned} T(t) &= B^f(t) + B_2(t) + B^b(t) \\ &= \int_0^t R_1(\alpha)d\alpha - \int_0^t R^b(\alpha)d\alpha \ . \end{aligned} \tag{5}$$

The first server serves traffic from the first buffer as fast as possible, but insures that $T(t) \leq K$ for all t. More specifically,

$$R_1(t) = \begin{cases} C - I_1(t) & \text{, if } B_1(t) > 0 \text{ and } T(t) < K \\ \min\{C - I_1(t), R_0(t)\} & \text{, if } B_1(t) = 0 \text{ and } T(t) < K \\ \min\{C - I_1(t), R^b(t)\} & \text{, if } B_1(t) > 0 \text{ and } T(t) = K \\ \min\{C - I_1(t), R_0(t), R^b(t)\} & \text{, if } B_1(t) = 0 \text{ and } T(t) = K \ . \end{cases} \tag{6}$$

Similarly, the second server serves traffic from the second buffer as fast as possible:

$$R_2(t) = \begin{cases} C - I_2(t) & , \text{ if } B_2(t) > 0 \\ \min\{C - I_2(t), R^f(t)\} & , \text{ if } B_2(t) = 0 \, . \end{cases} \tag{7}$$

The system evolution is completely determined by (1)-(7) and the functions R_0, I_1, I_2, R^f, and R^b. We characterize the network elements N^f and N^b by "service curves" S^f and S^b (defined below), respectively. Finally, we assume general "burstiness constraints" [1] on the cross-traffic, i.e. we assume that there exist non-negative non-decreasing functions b_i^{cross} such that for $i = 1, 2$ and all $s \leq t$ there holds

$$\int_s^t I_i(\alpha) d\alpha \leq b_i^{cross}(t - s) \, . \tag{8}$$

1.1 Discussion

The model described above has a number of applications. For example, it may model a single hop within an ATM network that uses credit based flow control. The network element N^f would model forward propagation delay in this case. It has been proposed that acknowledgements only be sent back periodically, rather than continuously, and so the network element N^b could model a combination of backward propagation delay and jitter caused by accumulation of acknowledgements until a burst of acknowledgements is sent back.

Another possible application is the situation in which *end-to-end* window flow control is applied across multiple hops. In this case, N^f would model a combination of queueing and propagation delay suffered in the forward path to the destination. Similarly, N^b would model a combination of queueing and propagation delay in the backward path for acknowledgements returning to the first server.

Alternatively, the model could describe the flow of data across a protocol layer within a host, whereby processes that feed a buffer enter a blocked state when the buffer reaches capacity, and the processes provide bursty service due to multi-tasking within an operating system.

More generally, the model is of interest in manufacturing networks where window flow control is induced naturally by limited storage space for parts being serviced by a sequence of machines, e.g. see [8].

The model described above is a generalization of a model previously proposed in [3], where N^f and N^b were not present (i.e. $R^f \equiv R_1$ and $R^b \equiv R_2$), $b_i^{cross}(x) = \sigma_i + \rho_i x$ (affine burstiness constraints on cross-traffic), and $B_1(0) = \infty$ (infinite supply of packets at the source). For this special case, a lower bound on the throughput of the system, $\liminf_{t\to\infty}(1/t) \int_0^t R_2$, was derived using a "Lyapunov function approach." Although this approach yielded tight results for two servers in tandem, it is difficult to generalize for more than two servers in tandem (see Figure 3) since finding an appropriate Lyapunov function is problematic. Instead, in [3], an aggregation and decomposition technique was used to recursively reduce the analysis of several servers in tandem to the two server case.

In this paper, we use the concept of "service curves" [2] to analyze the system described above. This allows us to easily extend the analysis for more than two servers in tandem, to incorporate "propagation" delays, and to consider general burstiness constraints on the cross-traffic.

Before analyzing the model above, we define service curves and present some new interpretations relating to linear system theory. In Section 3, we present our main results on the two server system discussed in this section. In Section 4, we apply the results of Section 3 to the case of several servers in tandem.

2 Service Curves

Consider a system with entering and exiting traffic described by the rate functions R_{in} and R_{out}. The amount of data stored in the system at time $t \geq 0$ is $B(t) = \int_0^t (R_{in}(\alpha) - R_{out}(\alpha))d\alpha$, where we assume $B(0) = 0$. Suppose S is a given non-negative function. To simplify the notation, assume without loss of generality that $S(x) = 0$ for all $x \leq 0$. Building upon the results in [9], the following definition was essentially proposed in [2] – it is adapted here to the continuous time case.

Definition 1. (Strict Service Curve Guarantee). *A system is said to strictly guarantee the service curve S if for all t, there exists $s \leq t$ with $B(s) = 0$ and $\int_s^t R_{out}(\alpha)d\alpha \geq S(t-s)$.*

The adjective "strict" was not used in [2]. It is added here to differentiate it from a slightly weaker service guarantee which we now introduce:

Definition 2. (Service Curve Guarantee). *A system is said to guarantee the service curve S if for all $t \geq 0$, there exists $s \leq t$ such that $\int_0^t R_{out}(\alpha)d\alpha - \int_0^s R_{in}(\alpha)d\alpha \geq S(t-s)$.*

Note that if $B(s) = 0$, then $\int_0^s R_{in}(\alpha)d\alpha = \int_0^s R_{out}(\alpha)d\alpha$, so that a strict service curve guarantee implies a service curve guarantee. The converse is not necessarily true.

In passing, we note that [10] a (σ, ρ) ("leaky bucket") regulator guarantees the service curve $S_{reg}(x) = \sigma + \rho x$, $x > 0$.

Given two functions F and G defined on the non-negative reals, define the convolution (over the "min-plus algebra") of F and G, written $F * G$, as

$$F * G(x) = \min\{F(x_1) + G(x_2) : x_1 \geq 0, x_2 \geq 0, x_1 + x_2 = x\} ,$$

where the minimum is replaced by an infimum if necessary. It is easy to verify that the convolution operation is commutative and associative, and that it distributes over the minimum operation. It is straightforward to verify that the following is in fact equivalent to Definition 2.

Alternative Definition 2. (Service Curve Guarantee). *A system is said to guarantee the service curve S if for all $t \geq 0$ there holds $\int_0^t R_{out}(\alpha)d\alpha \geq S(t) * \int_0^t R_{in}(\alpha)d\alpha$.*

Define the "impulse function" $\hat{\delta}(x) = 0$ if $x \leq 0$, and $\hat{\delta}(x) = +\infty$ if $x > 0$. Note that for any function F, $F * \hat{\delta}(x) = F(x)$. It is interesting to note then that a service curve is the impulse response of the network element, in some sense, under the min-plus algebra.

In [2], it is shown that if a system strictly guarantees a service curve, then bounds on delay, buffer size, and burstiness of the output traffic are easily derived, assuming burstiness constraints on the traffic arriving to the system. It turns out that identical bounds hold if the service curve guarantee is not necessarily strict. The proofs are almost identical – see [10] for the discrete time case. For completeness, we include these results below. The virtual delay at time t, $D(t)$, is defined as

$$D(t) = \min\{\Delta : \Delta \geq 0 \text{ and } \int_0^{t+\Delta} R_{out}(\alpha)d\alpha \geq \int_0^t R_{in}(\alpha)d\alpha\} .$$

Theorem A. [2] *Assume that* $\int_s^t R_{in}(\alpha)d\alpha \leq b(t-s)$ *for all* $s \leq t$. *Suppose a system guarantees a service curve of* S *(not necessarily strict). Then*
(a) (Buffer Requirements) There holds for all t

$$B(t) \leq \max_{\alpha \geq 0}\{[b(\alpha) - S(\alpha)]^+\} .$$

(b) (Bound on Delay) There holds for all t

$$D(t) \leq \max_{\alpha \geq 0}\{\min\{\Delta : \Delta \geq 0 \text{ and } b(\alpha) \leq S(\alpha + \Delta)\}\} .$$

(c) (Output Burstiness) For all $s \leq t$ *there holds* $\int_s^t R_{out}(\alpha)d\alpha \leq b_{out}(t-s)$, *where*

$$b_{out}(x) = \max_{\Delta \geq 0}\{b(x+\Delta) - S(\Delta)\} .$$

Theorem B. [2] (Convolution Theorem) *Consider traffic flowing through a system consisting of* n *subsystems in tandem, where the* i^{th} *subsystem guarantees the service curve* S_i *(not necessarily strict). Then the system as a whole guarantees the service curve* $S_{net} = S_1 * S_2 * \cdots * S_n$.

Given the analogy to linear filtering theory, it is natural to ask if there is a concept of a transform domain. Given a function F defined on the non-negative reals, define the "concave conjugate" of F, $\cap^* F$, as

$$\cap^* F(\rho) = \inf_{x:x \geq 0}\{\rho x - F(x)\} \quad , \rho \geq 0 .$$

Note that if F is concave, then $\cap^*[\cap^* F] = F$. Similarly, define the "convex conjugate" of F, $\cup^* F$, as

$$\cup^* F(\rho) = \sup_{x:x \geq 0}\{\rho x - F(x)\} \quad , \rho \geq 0 ,$$

and note that $\cup^*[\cup^* F] = F$ if F is convex. It is straightforward to verify that for two functions F and G we have

$$\cup^*[F * G](\rho) = \cup^* F(\rho) + \cup^* G(\rho) ,$$

which is the analogue of the convolution property for Fourier transforms. It is also interesting to study the mapping from b to b_{out} in Theorem A, part (c). It is easy to verify that the function ρx is an *eigenfunction* of this mapping, in some sense. Furthermore it is not difficult to show that

$$-\cap^*[b_{out}](\rho) \leq -\cap^*[b](\rho) + \cup^*[S](\rho) .$$

3 Service Curve for Window Flow Control

We return to the system illustrated in Figure 1. We will assume that network elements N^f and N^b guarantee service curves S^f and S^b respectively (not necessarily strict).

3.1 Service Curve for Window Flow Control

We focus on finding a service curve for the first buffer and server. Given any t, let $s^* = \max\{s : s \leq t \text{ and } B_2(s) = 0\}$. From (7), it follows that $R_2(s) = C - I_2(s)$ for $s \in (s^*, t)$, and therefore from (8) we have

$$\int_{s^*}^{t} R_2(\alpha)d\alpha \geq C(t - s^*) - b_2^{cross}(t - s^*) .$$

Since $R_2 \geq 0$, we have

$$\int_{s^*}^{t} R_2(\alpha)d\alpha \geq [C(t - s^*) - b_2^{cross}(t - s^*)]^+ .$$

Noting that $B_2(s^*) = 0$, it follows that the subsystem consisting of the second buffer and second server strictly guarantees the service curve S_2, where

$$S_2(x) = [Cx - b_2^{cross}(x)]^+ . \tag{9}$$

Note, however, that the subsystem consisting of the first buffer and first server does *not* necessarily guarantee the service curve $\hat{S}_1$ defined as

$$\hat{S}_1(x) = [Cx - b_1^{cross}(x)]^+ , \tag{10}$$

since the window flow control protocol may inhibit such a service guarantee. The next theorem identifies a service curve that *is* guaranteed by the subsystem consisting of the first buffer and first server.

Before stating the theorem, we introduce some convenient notation. Given a function G, and a positive integer n, let $G^{(n)}$ be the n-fold convolution of G with itself, i.e.

$$G^{(n)}(x) = \underbrace{G * G * \cdots * G}_{n}(x) .$$

In the case where $n = 0$, we define $G^{(0)}(x) = \hat{\delta}(x)$.

Theorem 1 (Service Curve for Window Flow Control) *Suppose that the subsystem consisting of the second buffer and second server guarantees the service curve S_2, and that network elements N^f and N^b guarantee the service curves S^f and S^b, respectively. Define $S_{loop}(x) = \hat{S}_1 * S^f * S_2 * S^b(x)$. Then the subsystem consisting of the first buffer and first server strictly guarantees the service curve S_1, where*

$$S_1(x) = \min_{m \in Z^+} \{\hat{S}_1 * S_{loop}^{(m)}(x) + mK\} , \tag{11}$$

and Z^+ is the set of non-negative integers.

Remark: It is interesting to note that that the service curve S_1 given in Theorem 1 is the "impulse response" of the linear feedback system depicted in Figure 2, under the "min-plus" algebra.

To prove Theorem 1 , we will use the lemma below.

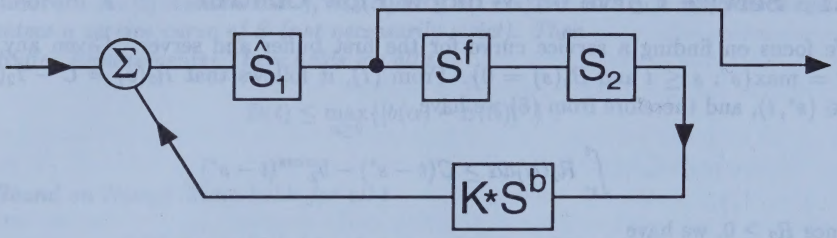


Figure 2: Window Flow Control as a Linear Feedback System.

Lemma 1 *For any fixed t, under the hypothesis of Theorem 1, there exists a finite sequence of intervals $(u_{n+1}, t_n), (t_n, u_n), \cdots, (u_1, t_0)$, with $t_0 = t$, such that $B_1(u_{n+1}) = 0$ and:*

$$\int_{u_{i+1}}^{t_i} R_1(\alpha)d\alpha \geq \hat{S}_1(t_i - u_{i+1}) \quad \forall i = 0, .., n \tag{12}$$

$$\int_{t_{i+1}}^{u_{i+1}} R_1(\alpha)d\alpha \geq S^f * S_2 * S^b(u_{i+1} - t_{i+1}) + K \quad \forall i = 0, .., n-1 \ . \tag{13}$$

Furthermore, the lengths of these intervals satisfy the constraints:

$$t_i - u_{i+1} \geq 0 \ , \quad i = 0, .., n \tag{14}$$

$$u_{i+1} - t_{i+1} \geq K/C \ , \quad i = 0, .., n-1 \ . \tag{15}$$

Proof of Lemma 1: Fix $t \geq 0$ and set $t_0 = t$. We define u_{i+1} in terms of t_i:

$$u_{i+1} = \max\{\max\{s : s \leq t_i, B_1(s) = 0\}, \max\{u : u \leq t_i, T(u) = K\}\} \ . \tag{16}$$

Note that $T(s) < K$ and $B_1(s) > 0 \ \forall s \in (u_{i+1}, t_i)$ and hence using (6) and (8) we have

$$\begin{aligned} \int_{u_{i+1}}^{t_i} R_1(\alpha)d\alpha &= C(t_i - u_{i+1}) - \int_{u_{i+1}}^{t_i} I_1(\alpha)d\alpha \\ &\geq C(t_i - u_{i+1}) - b_1^{cross}(t_i - u_{i+1}) \ . \end{aligned}$$

Remembering that $\int_{u_{i+1}}^{t_i} R_1(\alpha)d\alpha \geq 0$, and using the definition in (10), this implies

$$\int_{u_{i+1}}^{t_i} R_1(\alpha)d\alpha \geq \hat{S}_1(t_i - u_{i+1}) \ ,$$

which proves (12). If $B_1(u_{i+1}) = 0$ then set $n = i$.

Otherwise, if $B_1(u_{i+1}) > 0$, then note that $T(u_{i+1}) = K$. In this case, we will define t_{i+1} in terms of u_{i+1} as follows. By the convolution theorem (Theorem B), the system consisting of the series cascade of N^f, the second buffer and second server, and N^b guarantees a service curve of $S^f * S_2 * S^b(x)$. Thus, there exists $t_{i+1} \leq u_{i+1}$ such that

$$\int_0^{u_{i+1}} R^b(\alpha)d\alpha - \int_0^{t_{i+1}} R_1(\alpha)d\alpha \geq S^f * S_2 * S^b(u_{i+1} - t_{i+1}) \ . \tag{17}$$

Using equations (5) and (17), we get

$$\begin{aligned} K &= T(u_{i+1}) \\ &= \int_0^{u_{i+1}} R_1(\alpha)d\alpha - \int_0^{u_{i+1}} R^b(\alpha)d\alpha \\ &\leq \int_0^{u_{i+1}} R_1(\alpha)d\alpha - S^f * S_2 * S^b(u_{i+1} - t_{i+1}) - \int_0^{t_{i+1}} R_1(\alpha)d\alpha \ . \end{aligned} \tag{18}$$

This proves (13).

By construction, we have $t_i - u_{i+1} \geq 0$, which is (14). Since service curves are non-negative and R_1 is bounded above by C, we get from (18)

$$K \leq \int_{t_{i+1}}^{u_{i+1}} R_1(\alpha)d\alpha \leq C(u_{i+1} - t_{i+1}) \ , \tag{19}$$

which implies (15). By construction, $t_{i+1} \geq u_{i+2}$. Hence by adding the non-negative quantity $C(t_{i+1} - u_{i+2})$ to the right side of (19), we get $u_{i+1} - u_{i+2} \geq K/C > 0$. Thus, since the system is completely empty at time 0 and t is finite, the recursion must end in a finite number of iterations, i.e. $B_1(u_{i+1}) = 0$ for some finite i. Thus, the construction results in a finite sequence of intervals as claimed in the lemma.

◊

Proof of Theorem 1 : Fix any $t > 0$. Invoking Lemma 1, we have

$$\begin{aligned} \int_{u_{n+1}}^{t} R_1(\alpha)d\alpha &= \sum_{i=0}^{n} \int_{u_{i+1}}^{t_i} R_1(\alpha)d\alpha + \sum_{i=0}^{n-1} \int_{t_{i+1}}^{u_{i+1}} R_1(\alpha)d\alpha \\ &\geq \sum_{i=0}^{n} \hat{S}_1(t_i - u_{i+1}) + \sum_{i=0}^{n-1} \{S^f * S_2 * S^b(u_{i+1} - t_{i+1}) + K\} \\ &= \hat{S}_1(t_n - u_{n+1}) + \sum_{i=0}^{n-1} \{\hat{S}_1(t_i - u_{i+1}) + S^f * S_2 * S^b(u_{i+1} - t_{i+1}) + K\} \\ &\geq \hat{S}_1(t_n - u_{n+1}) + \sum_{i=0}^{n-1} \{S_{loop}(t_i - t_{i+1}) + K\} \\ &\geq \hat{S}_1 * S_{loop}^{(n)}(t - u_{n+1}) + nK \\ &\geq S_1(t - u_{n+1}) \ . \end{aligned}$$

Noting that $B_1(u_{n+1}) = 0$, this completes the proof.

◊

Suppose that the delay in network element N^f is upper bounded by τ_f, and that the delay in N^b is upper bounded by τ_b. This would happen if N^f and N^b represent propagation delay. In this case, it is easy to show that for $d = f$ or $d = b$, network element N^d guarantees the service curve S^d, where $S^d(x) = \hat{\delta}(x - \tau_d)$. We use this fact in the next corollary, which considers affine burstiness constraints on the cross-traffic.

Corollary 1 *Suppose $b_1^{cross}(x) = \sigma_a + \rho_a x$, $b_2^{cross}(x) = \sigma_b + \rho_b x$, and the delay through N^f and N^b are bounded by τ_f and τ_b, respectively. Define $\Delta = \tau_f + \tau_b + [\sigma_a/(C - \rho_a)] + [\sigma_b/(C - \rho_b)]$, and $\rho = \max\{\rho_a, \rho_b\}$. Then the system consisting of the first buffer and first server strictly guarantees a service curve of $\bar{S}_1(x)$ where*

$$\bar{S}_1(x) = \min_{m \in Z^+} Q_1^m(x) \ , \tag{20}$$

$$Q_1^0(x) = (C-\rho_a)(x - \frac{\sigma_a}{C-\rho_a})^+ = \hat{S}_1(x) \ , \tag{21}$$

and for $m \geq 1$*:*

$$Q_1^m(x) = (C-\rho)(x - m\Delta - \frac{\sigma_a}{C-\rho_a})^+ + mK \ . \tag{22}$$

The next corollary assesses the impact of the window size K. Define

$$K^* = \begin{cases} (C-\rho)(\tau_f+\tau_b) + \sigma_a + \frac{C-\rho_a}{C-\rho_b}\sigma_b & \text{, if } \rho_a > \rho_b \\ (C-\rho)(\tau_f+\tau_b) + \frac{C-\rho_b}{C-\rho_a}\sigma_a + \sigma_b & \text{, otherwise.} \end{cases} \tag{23}$$

Corollary 2 *Suppose* $b_1^{cross}(x) = \sigma_a + \rho_a x$ *and* $b_2^{cross}(x) = \sigma_b + \rho_b x$*, and the delay through* N^f *and* N^b *is upper bounded by* τ_f *and* τ_b *respectively. Then the system consisting of the first buffer and first server strictly guarantees a service curve of* $\bar{S}_1(x)$*, where if* $K \geq K^*$ *then*

$$\bar{S}_1(x) = \begin{cases} Q_1^0(x) = \hat{S}_1(x) & \text{, if } \rho_a > \rho_b \\ \min\{Q_1^0(x), Q_1^1(x)\} & \text{, otherwise ,} \end{cases} \tag{24}$$

and if $K < K^*$ *then*

$$\liminf_{x\to\infty} \frac{\bar{S}_1(x)}{x} = \frac{K}{\Delta} \ . \tag{25}$$

Remark: Corollary 2 implies that a window size equal to the sum of the burstiness of the cross-traffic at the first server (σ_a), the burstiness of the cross-traffic at the second server (σ_b), and the user bandwidth delay product ($(C-\rho)(\tau_f+\tau_b)$) is sufficient to guarantee the maximum guaranteed throughput of $C-\rho$.

3.2 Buffer Requirements for Window Flow Control

If the input stream R_0 is characterized by a burstiness constraint, an upper bound on $B_1(t)$ can be derived from Theorem A and Theorem 1. An obvious upper bound for $B_2(t)$ is the window size K, which can be achieved if the delay through network elements N^f and N^b is allowed to be zero. However, it is sometimes possible to derive an upper bound on $B_2(t)$ which is smaller than the window size K, even under no assumptions on R_0.

To illustrate this, we now consider the case where N^f and N^B represent *constant* delays of τ_f and τ_b, respectively. We will derive an upper bound on $B_2(t)$ using two facts. First, note that in this case we have

$$T(t-\tau_f) = \int_0^{t-\tau_f} R_1(\alpha)d\alpha - \int_0^{t-(\tau_f+\tau_b)} R_2(\alpha)d\alpha \ \leq \ K \ .$$

Thus,

$$\begin{aligned} B_2(t) &= \int_0^{t-\tau_f} R_1(\alpha)d\alpha - \int_0^t R_2(\alpha)d\alpha \\ &\leq \{K + \int_0^{t-(\tau_f+\tau_b)} R_2(\alpha)d\alpha\} - \int_0^t R_2(\alpha)d\alpha \\ &= K - \int_{t-(\tau_f+\tau_b)}^t R_2(\alpha)d\alpha \ . \end{aligned} \tag{26}$$

Second, in this case, the time elapsed from the time a piece of traffic leaves the first server until the acknowledgement for it returns to the first server is at least $\tau_f + \tau_b$. Thus, for any s and t such that $0 \leq t - s \leq \tau_f + \tau_b$ we have

$$\int_s^t R_1(\alpha)d\alpha \leq K \ . \tag{27}$$

Since R_1 is bounded above by the capacity of the server C, from (27) it follows that for any s and t such that $0 \leq t - s \leq \tau_f + \tau_b$ we have

$$\int_s^t R_1(\alpha)d\alpha \leq \min\{C(t-s), K\} \ . \tag{28}$$

Both inequalities (26) and (27) are due to [6].

Theorem 2 *Suppose that N^f and N^b represent constant delays τ_f and τ_b, respectively, and that the cross-traffic I_2 at the second server satisfies the burstiness constraint (8). Then the amount of traffic $B_2(t)$ in the second buffer satisfies*

$$B_2(t) \leq \max\{B_2^{\phi_1}, B_2^{\phi_2}\} \ , \tag{29}$$

where

$$\begin{aligned} B_2^{\phi_1} &= \max_{x:0\leq x\leq \tau_f+\tau_b} \{\min\{Cx, K\} - [Cx - b_2^{cross}(x)]^+\} \\ B_2^{\phi_2} &= K - [C(\tau_f+\tau_b) - b_2^{cross}(\tau_f+\tau_b)]^+ \ . \end{aligned}$$

Proof Fix t and define $u = \max\{s : s \leq t, B_2(s) = 0\}$. If $t - u > \tau_f + \tau_b$ then $B_2(s) > 0$ for all $s \in (t - \tau_f - \tau_b, t)$ and hence

$$\begin{aligned} \int_{t-\tau_f-\tau_b}^t R_2(\alpha)d\alpha &= C(\tau_f+\tau_b) - \int_{t-\tau_f-\tau_b}^t I_2(\alpha)d\alpha \\ &\geq [C(\tau_f+\tau_b) - b_2^{cross}(\tau_f+\tau_b)]^+ \ . \end{aligned} \tag{30}$$

Thus, in this case it follows from (26) that $B_2(t) \leq B_2^{\phi_2}$.

If $t - u \leq \tau_f + \tau_b$ then using (28) it follows that

$$\begin{aligned} B_2(t) &= \int_u^t R^f(\alpha)d\alpha - \int_u^t R_2(\alpha)d\alpha \\ &= \int_{u-\tau_f}^{t-\tau_f} R_1(\alpha)d\alpha - \int_u^t [C - I_2(\alpha)]d\alpha \\ &\leq \min\{C(t-u), K\} - [C(t-u) - b_2^{cross}(t-u)]^+ \\ &\leq B_2^{\phi_1} \ . \end{aligned}$$

◇

As an example, suppose that $b_2^{cross}(x) = \sigma_b + \rho_b x$, and $\sigma_b/(C - \rho_b) \leq K/C \leq \tau_f + \tau_b$. In this case Theorem 2 yields that

$$B_2(t) \leq \sigma_b + \rho_b(K/C) \ .$$

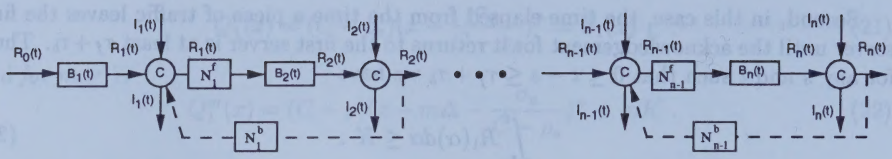


Figure 3: Several Servers in Tandem.

As claimed, the upper bound above may be considerably smaller than the window size K. Smaller upper bounds on $B_2(t)$ can be obtained if R_1 is more strongly regulated. One way to achieve this is to assume a *lower* bound on the cross-traffic at the first server. In this case, if there is insufficient cross-traffic at the first server to achieve this, the first server could "pretend" there is additional cross-traffic. Perhaps a better approach is to have the first server directly control the burstiness of R_1. This is a topic for future research.

4 Several Servers in Tandem

We now investigate the several server case as shown in Figure 3 where we have n cascaded buffer-server pairs. The model is analogous to the two server case and precise definitions are omitted here for brevity.

In order to analyze this multiserver system, we may apply Theorem 1 recursively as follows. Given a burstiness constraint for the cross-traffic at the n^{th} server, a service curve S_n strictly guaranteed by the n^{th} buffer-server pair is implied, analogous to (9). Theorem 1 then implies that a service curve S_{n-1} is strictly guaranteed by the $(n-1)^{th}$ buffer-server pair. This process is repeated until a strict service curve guarantee of S_j is derived for the j^{th} buffer-server pair for all j. The end-to-end service curve for the entire system is then $(S_1 * S_2 * \cdots * S_n) * (S_1^f * S_2^f * \cdots * S_n^f)(x)$ by Theorem B. The following corollary follows by using this method.

Corollary 3 *Consider the tandem configuration of n buffer-server pairs as illustrated in Figure 3, where network elements N_j^f and N_j^b have maximum delays τ_j^f and τ_j^b, respectively. Suppose $b_i^{cross}(x) = \sigma_i + \rho_i x$ for all $i = 1, .., n$, where $\rho = \rho_1 = \rho_2 = ... = \rho_n$.*

If $K_j \geq \sigma_j + \sigma_{j+1} + (C-\rho)(\tau_b^j + \tau_f^j) \quad \forall j = 1, ..n-1$, then the system consisting of the j^{th} buffer and the j^{th} server strictly guarantees a service curve of

$$S_j(x) \;=\; [-\sigma_j + (C-\rho)x]^+ \quad \forall \quad j = 1, ..., n \; .$$

Furthermore, the entire system guarantees a service curve of $S_{total}(x) = (C-\rho)(x - \sum_{j=1}^{n} \frac{\sigma_j}{C-\rho} - \sum_{j=1}^{n-1} \tau_f^j)^+$. If R_0 is such that $\int_s^t R_0(\alpha)d\alpha \leq \sigma_0 + \rho_0(t-s)$ for all $s \leq t$, where $\rho_0 \leq C - \rho$, then the total end-to-end delay is bounded above by D_{total}, where

$$D_{total} = \frac{\sum_{j=0}^{n} \sigma_j}{C-\rho} + \sum_{j=1}^{n-1} \tau_f^j \; . \tag{31}$$

The results in Corollary 3 improve previously reported bounds, where we showed that $\sum_{j=1}^{n-1} K_j = O(n^2)$ was sufficient for a guaranteed throughput of $C - \rho$ [3] and for a maximum total end-to-end delay bounded by $O(n^2)$ [4].

References

[1] R. Cruz, "A Calculus for Network Delay, Part I: Network Elements in Isolation," *IEEE Transactions on Information Theory*, Vol. 37, No. 1, Jan. 1991.

[2] R. Cruz, "Quality of service guarantees in virtual circuit switched networks," *IEEE Journal on Selected Areas in Communications*, Vol. 13, No. 6, Aug. 1995.

[3] R. Cruz, "Guaranteed throughput for window flow control in a multi-tasking environment," Proceedings of the 1991 Conference on Information Sciences and Systems, John Hopkins University, March 1991.

[4] R. Cruz, 'Tandem queues with finite buffers in a multi-tasking environment: performance guarantees via rate control," Proceedings of the 1992 Conference on Information Sciences and Systems, Princeton University, March 1992.

[5] E. L. Hahne, "Round robin scheduling for fair flow control in data communications networks," *IEEE JSAC*, vol. 9, no. 7, Sept. 1991, pp. 1024-1039.

[6] S. Khorsandi and A. Leon-Garcia, "Robust non-probabilistic bounds for delay and throughput in credit-based flow control," Proceedings IEEE INFOCOM '96, vol. 2, pp 577-84.

[7] H. T. Kung, T. Blackwell, and A. Chapman, "Credit update protocol for flow controlled ATM networks: statistical multiplexing and adaptive credit allocation," Proceedings of SIGCOMM'94, pp. 101-1115.

[8] S. M. Meerkov and F. Top, "Analysis and synthesis of asymptotically reliable serial production lines," Technical Report CGR-24, Dept. of EECS, University of Michigan, Nov. 1989.

[9] A. K. Parekh and R. G. Gallager, "A generalized processor sharring approach to flow control in integrated services networks: the single node case," *IEEE/ACM Trans. Networking,"* vol. 1, no. 3, June 1993, pp. 344-57.

[10] H. Sariowan, "A service-curve approach to performance guarantees in integrated-service networks," Ph.D. thesis, Dept. of Electrical & Computer Engineering, UCSD, June 1996.

Fast Algorithms for Measurement-Based Traffic Modeling *

Hao Che *San-qi Li*
Department of Electrical and Computer Engineering
University of Texas at Austin
Austin, Texas 78712

October 1, 1996

Abstract

This paper develops fast algorithms for construction of circulant modulated rate process to match with two primary traffic statistical functions: rate distribution $f(x)$ and autocorrelation $R(\tau)$. Using existing modeling techniques, $f(x)$ has to be limited to certain forms such as Gaussian or binomial; $R(\tau)$ can only consist of one or two exponential terms which are often real exponentials rather than complex. In reality, these two functions are collective from real traffic traces and generally expressed in much complicated form. Our emphasis here is placed on the algorithmic design for matching complicated $R(\tau)$ in traffic modeling. The typical CPU time for the traffic modeling with $R(\tau)$ consisting of five or six complex exponential terms is found in the range of a few minutes by the proposed algorithms. Our study further shows an excellent agreement between original traffic traces and sequences generated by the matched analytical model.

1 Introduction

Many studies indicate the importance of high burstiness and strong correlation nature of multimedia traffic to network control and resource management design. Such traffic dynamics in statistical measurement are mainly captured by the collective rate distribution function $f(x)$ and autocorrelation function $R(\tau)$ of the arrival rate process. Real traffic measurement shows that both $f(x)$ and $R(\tau)$ are generally expressed in complicated form. The central task of *measurement-based* traffic modeling is to develop algorithms for construction of random process models, whose statistics match with the collective $f(x)$ and $R(\tau)$. This is difficult especially if the models should fit into feasible analytical queueing solutions for network performance evaluation.

*The research reported here was supported by NSF under grant NCR-9015757 and by Texas Advanced Research Program under grant TARP-33.

Most queueing analyses have used two-state Markov chain modulated process as a basic element for construction of multimedia traffic. Using this modeling technique, $R(\tau)$ has to be in the form $\sum_k \psi_k e^{\lambda_k \tau}$ with real ψ_k and real λ_k; $f(x)$ is limited to a convolved binomial function. They are certainly insufficient to capture the diversed traffic correlation and burstiness behavior. In [1, 2], *Li* and *Hwang* proposed a structure of circulant modulated Poisson process (CMPP), whose $R(\tau)$ has the form $\sum_k \psi_k e^{\lambda_k \tau}$ with real ψ_k but complex λ_k and whose cumulative function $F(x) = \int_0^x f(u)du$ is expressed as a piece-wise step function. It is obvious that the sum of complex exponentials represent a much wider class of correlation functions than the sum of real exponentials.

One may question the validity of using circulant for traffic modeling. A comprehensive, blind numerical study in [2, 3] compared queueing/loss-rate solutions between vastly different circulant and Markov chain as well as nonMarkovian process including auto regressive moving average model (ARMA), nonlinearized ARMA model for nonGaussian distribution, and correlated batch arrival process with nonexponential interarrival distribution. All the comparison results showed excellent agreement,provided matching with identical $f(x)$ and $R(\tau)$.

The measurement-based construction of CMPP consists of three steps [2]. The first step is the identification of ψ_k's and λ_k's to match a collective correlation sequence, which is formulated as a standard nonlinear programming problem. In this paper we develop a heuristic algorithm by taking the combination of the Prony algorithm with the non-negative least squares (NNLS) method. The second step is the formation of a circulant transition rate matrix whose eigenvalues must contain all the λ_k's of $R(\tau)$, which is originally an inverse eigenvalue problem but transformed to an index search problem in [2]. The major contribution of this paper is the development of a fast index search algorithm for increased number of λ_k's. The third step of the construction is the design of input rate vector for CMPP to optimally match a collective rate distribution $F(x)$, which is formulated as a minimization problem and effectively solved by the Nelder-Meade simplex search method in [2].

Other than using the superposition of two-state Markov chains, a few works are available for measurement-based traffic modeling to match with collective $R(\tau)$ and $f(x)$. In [4], a multi-state Markov chain is constructed to model a MPEG video, but its correlation function still contains a single real exponential as a two-state Markov chain. In [5], *Jagerman* and *Melamed* proposed a measurement-based modeling technique called transform expand sample (TES), which is mainly suitable for computer simulation.

The paper is organized as follows. Section 2 provides the background knowledge of the algorithmic design for CMPP construction. Section 3 describes a heuristic fast algorithm for the identification of λ_k's and ψ_k's to match with a given correlation sequence. Section 4, which is the major part of this paper, develops a fast index search algorithm for the construction of a circulant transition rate matrix whose eigenvalues contain all the λ_k's. Section 5 provides modeling examples based on real traffic traces to show the potential of the algorithm.

2 Background

Our objective here is the construction of a CMPP to statistically match with a stationary random arrival process $a(t)$. Denote the correlation and cumulative rate distribution functions of $a(t)$ by $R(\tau)$ and $F(x)$ and that of CMPP by $R_c(\tau)$ and $F_c(x)$. The algorithmic design is to achieve

$$R_c(\tau) \approx R(\tau), \qquad F_c(x) \approx F(x).$$

In reality, $R(\tau)$ is either expressed analytically or described by a correlation sequence collective from real traffic trace. For CMPP, $R_c(\tau)$ must be represented by

$$R_c(\tau) = \sum_{l=0}^{M} \psi_l e^{\lambda_l |\tau|} \tag{1}$$

with real nonnegative ψ_l and complex λ_l [2]. Define $\boldsymbol{\lambda} = [0, \lambda_1, \lambda_2, ..., \lambda_M]$ and $\boldsymbol{\psi} = [\psi_0, \psi_1, \psi_2, ..., \psi_M$ where M is the order of $R_c(\tau)$. The first step of our design is the identification of $\boldsymbol{\lambda}$ and $\boldsymbol{\psi}$ to match the original $R(\tau)$, i.e.,

$$\min_{\lambda_l, \psi_l} \int_0^\infty [R(\tau) - R_c(\tau)]^2 d\tau \qquad \text{s.t.} \qquad \psi_l \geq 0 \quad \forall l \tag{2}$$

which will be addressed in Section 3.

The next step is the construction of a circulant transition rate matrix $\mathbf{Q}$. Denote the first row of $\mathbf{Q}$ by vector $\boldsymbol{a} = [a_0, a_1, ..., a_{N-1}]$ for N-state circulant. Each row of $\mathbf{Q} = circ(\boldsymbol{a})$ is then formed by a forward shift permutation of the previous row. Define the eigenvalues of $\mathbf{Q}$ in vector form by $\boldsymbol{\lambda}_c = [\lambda_{c0}, \lambda_{c1}, ..., \lambda_{c(N-1)}]$. One can write

$$\boldsymbol{\lambda}_c = \sqrt{N} \boldsymbol{a} \mathbf{F}^\star. \tag{3}$$

$\mathbf{F}$ is a Fourier matrix with its (i,j)-th element given by $\frac{1}{\sqrt{N}} e^{-\frac{2\pi ij}{N}\sqrt{-1}}$. We have $\mathbf{F}^{-1} = \mathbf{F}^\star$ where $\mathbf{F}^\star$ is the conjugate transpose of $\mathbf{F}$. For CMPP, each eigenvalue contributes an exponential term to its correlation function,

$$R_c(\tau) = \sum_{k=0}^{N-1} \psi_{ck} e^{\lambda_{ck} |\tau|}. \tag{4}$$

Since the power vector $\boldsymbol{\psi}_c = [\psi_{c0}, \psi_{c1}, ..., \psi_{c(N-1)}]$ is assigned independent of $\boldsymbol{a}$ [2], the basic construction of $\boldsymbol{a}$ is to include $\boldsymbol{\lambda}$ in $\boldsymbol{\lambda}_c$. For the *wanted* eigenvalues, namely $\lambda_{cl} \in \boldsymbol{\lambda}, \forall l$, one can assign ψ_{cl} by their corresponding value in $\boldsymbol{\psi}$. For the remaining *unwanted* eigenvalues, namely $\lambda_{cl} \notin \boldsymbol{\lambda}, \forall l$, one can simply remove them by assigning $\psi_{cl} = 0$.

As a transition rate vector, $\boldsymbol{a}$ must be constructed under the nonnegative constraint $\boldsymbol{a} \geq 0$, $a_k \geq 0$ for $k = 1, 2, ..., N-1$ with $a_0 = -\sum_{k=1}^{N-1} a_k$, which substantially limits its eigenvalue space. Nevertheless, only a subset of $\boldsymbol{\lambda}_c$ needs to match with $\boldsymbol{\lambda}$. One can significantly expand the eigenvalue space of $\boldsymbol{\lambda}_c$ by increasing its dimension N, i.e., making $N >> M$, such that a solution $\boldsymbol{a}$ for $\boldsymbol{\lambda} \subset \boldsymbol{\lambda}_c$ is more likely to exist.

The construction of $\boldsymbol{a}$ from $\boldsymbol{\lambda}$ is therefore transformed into an index search problem, which is to find an m-dimension index of $\boldsymbol{\lambda}$ within the N-dimension index space of $\boldsymbol{\lambda}_c$ for solution $\boldsymbol{a}$ in (3) to satisfy the nonnegative constraint $\boldsymbol{a} \geq 0$. Note that the unwanted eigenvalues in $\boldsymbol{\lambda}_c$ are arbitrarily assigned under the nonnegative constraint. Such index search has the space complexity of N^M. For $N >> M$, any brute force search method will fail to apply, especially for large M and N. Developing a fast index search algorithm therefore becomes crucial to the success of the proposed modeling technique. This is achieved in Section 4.

Once $\boldsymbol{a}$ is fixed by $\boldsymbol{\lambda}$, the last step of our design is to find an input rate vector $\boldsymbol{\gamma}$ for $F_c(x) \approx F(x)$. This problem was formulated as a minimization problem [2], which was effectively solved using the Nelder-Meade simplex search method. The total computation time in this step is typically in the range of a few seconds on a workstation. This paper focuses on the algorithmic design of the first two steps.

3 Matching Correlation Function

This section developes a heuristic fast algorithm for identification of $\boldsymbol{\lambda}$ and $\boldsymbol{\psi}$ in correlation function matching. The traffic statistics in real measurement are often collective in the discrete

time domain using computer and digital signal processing technologies. Without loss of generality, the original correlation function $R(\tau)$ can be expressed in the discrete time domain by a correlation sequence

$$\boldsymbol{r} = [R(0), R(1), ..., R(L-1)].$$

Similarly, one can replace $R_c(\tau)$ in (1) by $R_c(n) = \sum_{l=0}^{M} \psi_l e^{\lambda_l n}$ for $0 \leq n < L$. Define

$$\boldsymbol{\phi}_l = [1, e^{\lambda_l}, e^{2\lambda_l}, ..., e^{(L-1)\lambda_l}]^T \qquad \text{for } 0 \leq l \leq M. \tag{5}$$

For the matching in the discrete domain, one can rewrite (2) as

$$\min_{\{\boldsymbol{\lambda}, \boldsymbol{\psi}\}} E_1 = ||\mathbf{r} - \sum_{l=0}^{M} \psi_l \boldsymbol{\phi}_l||^2, \qquad \text{s. t. } \boldsymbol{\psi} \geq 0 \tag{6}$$

where $||.||$ is the Euclidian norm.

This is a nonlinear least square (NLS) problem with constraints, i.e., a nonlinear programming problem, which is generally very difficult to solve since no efficient algorithms are available to date. Here we propose a heuristic algorithm which is composed of three steps. In the first step, we simply use the standard Prony method [8] to identify an intermediate solution $(\boldsymbol{\lambda}_{int}, \boldsymbol{\psi}_{int})$ without the real nonnegative constraint on $\boldsymbol{\psi}_{int}$. In the second step, the intermediate solution vector $\boldsymbol{\lambda}_{int}$ is much expanded to $\boldsymbol{\lambda}_{exp}$ with $\boldsymbol{\lambda}_{int} \subset \boldsymbol{\lambda}_{exp}$. The extra eigenvalues in $\boldsymbol{\lambda}_{exp}$ are generated through interpolation of the eigenvalues in $\boldsymbol{\lambda}_{int}$. It is based on the intuition that the eigenvalues of the constraint solution $\boldsymbol{\lambda}$ are likely in the neighborhood of the eigenvalues of the nonconstraint solution $\boldsymbol{\lambda}_{int}$. In other words, the correlation function captured by $(\boldsymbol{\lambda}_{int}, \boldsymbol{\psi}_{int})$, is likely to resemble that by $(\boldsymbol{\lambda}, \boldsymbol{\psi})$, since the former is formulated with less constraint than the latter. The purpose of the expansion is to approximately achieve $\boldsymbol{\lambda} \subset \boldsymbol{\lambda}_{exp}$. Various interpolation techniques can be used to expand $\boldsymbol{\lambda}_{int}$ to $\boldsymbol{\lambda}_{exp}$. The one used here is a simple linear interpolation in the complex domain among all the eigenvalues in $\boldsymbol{\lambda}_{int}$, which is found sufficient in most our applications. The last step is to find $\boldsymbol{\psi}_{exp}$ of the given $\boldsymbol{\lambda}_{exp}$ with the real nonnegative constraint of $\boldsymbol{\psi}_{exp} > 0$, which will minimize the difference between the original correlation sequence $\mathbf{r}$ and the correlation function of $(\boldsymbol{\lambda}_{exp}, \boldsymbol{\psi}_{exp})$. This is achieved by the well known non-negative least squares (NNLS) method [8]. Note that most power elements in $\boldsymbol{\psi}_{exp}$ will be zero since the order of the correlation function $(\boldsymbol{\lambda}_{exp}, \boldsymbol{\psi}_{exp})$ should not be significantly greater that the order of the correlation function $(\boldsymbol{\lambda}_{int}, \boldsymbol{\psi}_{int})$. Finally, the power vector $\boldsymbol{\psi}$ consists of the nonzero power elements of $\boldsymbol{\psi}_{exp}$; the eigenvalue vector $\boldsymbol{\lambda}$ consists of those eigenvalues in $\boldsymbol{\lambda}_{exp}$, whose corresponding power element in $\boldsymbol{\psi}_{exp}$ must be nonzero.

4 Index Search Algorithm

This section develops a fast index search algorithm (ISA) for construction of $\boldsymbol{a}$ from $\boldsymbol{\lambda}$, based on $\boldsymbol{\lambda}_c = \sqrt{N}\boldsymbol{a}\mathbf{F}^\star$ in (3) and $\boldsymbol{\lambda} \subset \boldsymbol{\lambda}_c$ subject to $\boldsymbol{a} \geq 0$. Subsection 4.1 formulates the problem. Subsection 4.2 presents the ISA algorithm. Subsection 4.3 provides examples to show the efficiency of the algorithm.

4.1 *Index Search Problem*

Since $\boldsymbol{a}$ is real, a complex λ_{cj} must be the conjugate of $\lambda_{c(N-j)}$ in $\boldsymbol{\lambda}_c$. In other words, the equation set by λ_{cj} in $\boldsymbol{\lambda}_c = \sqrt{N}\boldsymbol{a}\mathbf{F}^\star$ is equivalent to the equation set by $\lambda_{c(N-j)}$; only one complex eigenvalue in each conjugate pair needs to be taken into account in $\boldsymbol{\lambda}_c = \sqrt{N}\boldsymbol{a}\mathbf{F}^\star$. Further, since all the unwanted eigenvalues, $\lambda_{cj} \in \boldsymbol{\lambda}_c$ but $\notin \boldsymbol{\lambda}$, are arbitrarily assigned, they

shall be completely removed from $\boldsymbol{\lambda}_c = \sqrt{N}\boldsymbol{a}\mathbf{F}^\star$ for the derivation of $\boldsymbol{a}$. In consequence, one can rewrite $\boldsymbol{\lambda}_c = \sqrt{N}\boldsymbol{a}\mathbf{F}^\star$ as

$$\mathbf{A}(\boldsymbol{i}_m)\boldsymbol{a} = \boldsymbol{b}(\boldsymbol{i}_m) \qquad \text{and} \qquad \boldsymbol{a} \geq 0 \tag{7}$$

with $\boldsymbol{a} = [-a_0, a_1, ..., a_{N-1}]^T$, $\boldsymbol{i}_m = [i_1, i_2, ..., i_m]$ and

$$\boldsymbol{b}(\boldsymbol{i}_m) = [-\mathrm{Re}(\lambda_{i_1}), \mathrm{Im}(\lambda_{i_1}), ..., -\mathrm{Re}(\lambda_{i_m}), \mathrm{Im}(\lambda_{i_m})]^T,$$

$$\mathbf{A}(\boldsymbol{i}_m) = \begin{bmatrix} -1 & 1 & 1 & \cdots & 1 \\ 1 & -c_{i_1,1} & -c_{i_1,2} & \cdots & -c_{i_1,N-1} \\ 0 & s_{i_1,1} & s_{i_1,2} & \cdots & s_{i_1,N-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & -c_{i_m,1} & -c_{i_m,2} & \cdots & -c_{i_m,N-1} \\ 0 & s_{i_m,1} & s_{i_m,2} & \cdots & s_{i_m,N-1} \end{bmatrix}$$

where $c_{i,j} \stackrel{\text{def}}{=} \cos(2\pi ij/N)$ and $s_{i,j} \stackrel{\text{def}}{=} \sin(2\pi ij/N)$. m represents the number of distinctive, nonzero eigenvalues in $\boldsymbol{\lambda}$, where each pair of complex conjugate eigenvalues is counted as one distinctive eigenvalue. For $\boldsymbol{\lambda} \subset \boldsymbol{\lambda}_c$, the index vector $\boldsymbol{i}_m$ provides a specific index assignment of $\boldsymbol{\lambda}$ in $\boldsymbol{\lambda}_c$, i.e., $\boldsymbol{\lambda} = [\lambda_{i_1}, \lambda_{i_2}, ..., \lambda_{i_m}]$. A feasible solution $\boldsymbol{a}$ is obtained by identifying a specific $\boldsymbol{i}_m$ to satisfy (7).

Denote the set of entire index assignment space by $\boldsymbol{I}_m = \{\boldsymbol{i}_m : i_l \in \{1, 2, ..., N-1\}, \forall l \text{ for } i_l \neq i_j \text{ if } l \neq j\}$. The index space of $\boldsymbol{I}_m$ is then on the order of N^m. Our objective is to find an index vector $\boldsymbol{i}_m \in \boldsymbol{I}_m$ for a feasible solution $\boldsymbol{a}$ in (7). For convenience, we also denote the set of all feasible index assignments by $\boldsymbol{I}'_m$ with $\boldsymbol{i}_m \in \boldsymbol{I}'_m$ and $\boldsymbol{I}'_m \subset \boldsymbol{I}_m$. Typically, $\boldsymbol{I}'_m$ is a small fraction of $\boldsymbol{I}_m$.

Here we briefly discuss the ad-hoc scheme developed in [2] for solving (7) and explain the crucial importance of the proposed fast algorithm to the measurement-based modeling. In [2], (7) was formulated as a linear programming problem with an arbitrarily assigned objective function; the simplex method was employed to search for a feasible solution $\boldsymbol{a}$ at each selected $\boldsymbol{i}_m \in \boldsymbol{I}_m$. Since $\boldsymbol{i}_m$ is randomly selected at each time with incremental change of every index $i_l \in \boldsymbol{i}_m$, we call the scheme as random selection algorithm (RSA). The random search of this kind is found extremely ineffective for large N and m. Note that N has to be large for the expansion of solution space, which is typically in the range of several hundreds in our application. For instance, taking $N = 100$ at $m = 3$, a complete search of the index space $\boldsymbol{I}_3 = 10^6$ by RSA requires several hours of CPU time on SPARC workstation. The exponential expansion of the index space with m renders any brute force search algorithm to be virtually impractical for $m > 3$. In the RSA implementation, the number of allowable index searches at each given N was limited to a few hundreds, which only covers about 0.01% of the index space for $N = 100$ at $m = 3$. Since $\boldsymbol{I}'_m$ is only a small fraction of $\boldsymbol{I}_m$ such as 10^{-5} or less in most our experiments, the probability of finding a feasible $\boldsymbol{i}_m$ in $\boldsymbol{I}_m$ by RSA diminishes rapidly as m increases. Yet, confining $m \leq 3$ is certainly insufficient to capture the complex correlation behavior of multimedia traffic.

4.2 *Tree Search Algorithm*

Consider the index search problem of order n for $0 < n \leq m$. At each given n we have the notation of $\boldsymbol{i}_n$, $\boldsymbol{I}_n$ and $\boldsymbol{I}'_n$. Further define $\boldsymbol{i}_n = [\boldsymbol{i}_{n-1}, i_n]$, where $\boldsymbol{i}_{n-1}$ represents the first $(n-1)$ indices of $\boldsymbol{i}_n$. We introduce a new set $\boldsymbol{J}_{n-1} = \{\boldsymbol{i}_{n-1} : \boldsymbol{i}_n \in \boldsymbol{I}'_n\}$, i.e., $\boldsymbol{J}_{n-1}$ is the set of the index vectors which are composed of the first $(n-1)$ indices of $\boldsymbol{i}_n \in \boldsymbol{I}'_n$. The ISA algorithm is

developed on the basis of the following relation:

$$\boldsymbol{J}_n \subset \boldsymbol{I}'_n \qquad \forall n, \tag{8}$$

which is proved by contradiction. Consider an $\boldsymbol{i}_n \in \boldsymbol{I}'_n$, whose first $(n-1)$ elements are denoted by the corresponding $\boldsymbol{i}_{n-1} \in \boldsymbol{J}_{n-1}$. Suppose $\boldsymbol{i}_{n-1} \notin \boldsymbol{I}'_{n-1}$. By definition, $\boldsymbol{i}_n$ must be a feasible solution of (7). Since $\boldsymbol{i}_{n-1}$ represents the first $(n-1)$ elements of $\boldsymbol{i}_n$, it must also be a feasible solution of (7) with its last two equations removed. Yet , removing the two last equations in (7) is equivalent to formulating the index search problem of order $n-1$, whose solutions are denoted by $\boldsymbol{i}_{n-1} \in \boldsymbol{I}'_{n-1}$, which contradicts to the assumption $\boldsymbol{i}_{n-1} \notin \boldsymbol{I}'_{n-1}$. Hence, we must have $\boldsymbol{J}_{n-1} \subset \boldsymbol{I}'_{n-1}, \forall n$.

The relation simply states that for any $\boldsymbol{i}_n \in \boldsymbol{I}'_n$, we must have $\boldsymbol{i}_{n-1} \in \boldsymbol{I}'_{n-1}$ with $\boldsymbol{i}_n = [\boldsymbol{i}_{n-1}, i_n]$. Hence, starting at $n = 1$ we get $\boldsymbol{I}'_1$ with $\boldsymbol{i}_1 = [i_1]$. The solution $\boldsymbol{I}'_2$ can then be obtained by the search of index i_2 in $\boldsymbol{i}_2 = [\boldsymbol{i}_1, i_2]$ for $\boldsymbol{i}_1 \in \boldsymbol{I}'_1$. In general, the solution $\boldsymbol{I}'_n$ is obtained by the search of index i_n in $\boldsymbol{i}_n = [\boldsymbol{i}_{n-1}, i_n]$ for every $\boldsymbol{i}_{n-1} \in \boldsymbol{I}'_{n-1}$.

Since we are interested in just getting one feasible solution in $\boldsymbol{I}'_m$, the ISA is best implemented by tree search. Initially, we have $\boldsymbol{I}_0 = \emptyset$ as the root of the tree. Every branch at level 1 is then formed by $\boldsymbol{i}_1 \in \boldsymbol{I}'_1$. From $\boldsymbol{i}_n = [\boldsymbol{i}_{n-1}, i_n]$, it is clear that every branch at level n is formed by $\boldsymbol{i}_n \in \boldsymbol{I}'_n$, which is built on top of a branch at level $n-1$. The branch $\boldsymbol{i}_{n-1}$ terminates at level $n-1$ if $\boldsymbol{i}_n = [\boldsymbol{i}_{n-1}, i_n] \notin \boldsymbol{I}'_n, \forall i_n$.

The ISA can be stated as the following:

Input $\{m, N, \boldsymbol{\lambda}\}$;
$count = [0, 0, ..., 0]$;
$n = 0$;

for $i_n = count(n) + 1 : N - 1$
  $\boldsymbol{i}_n$=$\{\boldsymbol{i}_{n-1}, i_n\}$;
  Solve (7) by modified simplex (See Section 4.3) for $\boldsymbol{i}_n$;
  if feasible solution is found
    if $n = m$, output:*solution found*! and exit
    else $count(n) = i_n$; $n = n + 1$;
  else $count(n) = 0$; $n = n - 1$;

print: *No feasible solution found*!
end

The design of fast ISA algorithm is based on two principles: (a) to minimize the total number of searches and (b) to minimize the computation time of each search.

The structure of the ISA is well suited for the so called sensitivity analysis [7]. Recall that the search of i_n in $\boldsymbol{i}_n = [\boldsymbol{i}_{n-1}, i_n]$ on level n is performed at each given $\boldsymbol{i}_{n-1} \in \boldsymbol{I}'_{n-1}$. For $i_n \in \{1, 2, ..., N-1\}$, each $\boldsymbol{i}_{n-1}$ is applied at most $(N-1)$ times in the search of i_n before the selection of the next $\boldsymbol{i}_{n-1}$. Further, $\mathbf{A}(\boldsymbol{i}_n)\boldsymbol{a} = \boldsymbol{b}(\boldsymbol{i}_n)$ is equivalent to $\mathbf{A}(\boldsymbol{i}_{n-1})\boldsymbol{a} = \boldsymbol{b}(\boldsymbol{i}_{n-1})$ except with the addition of the last two equations which are associated with a newly selected eigenvalue. It implies that the output tableau of the simplex obtained for $\boldsymbol{i}_{n-1}$ on level $n-1$ can be directly used by the simplex for $\boldsymbol{i}_n$ on level n, which reduces the time complexity of each search by a factor proportional to n. Readers should refer to [6] for the detail formulation of this problem. Our comprehensive case study further indicates that the number of iterations required for $n \leq 8$ and $N \sim 100$ is on average less than 30, which grows very slowly with n. The sensitivity analysis therefore reduces the time complexity of ISA significantly.

m	3	4	5	6	7	8
cpu(m)(min.)	0.027	0.084	0.59	1.57	3.26	5.10
cpu(m)/cpu(m-1)		3	7	3	2	1.5

Table 1: Average CPU times measured over 100 randomly generated cases at $N = 101$

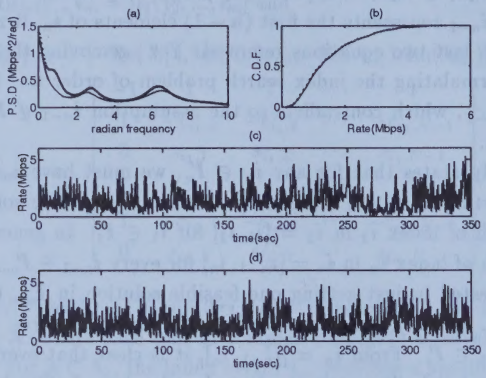


Figure 1: Comparison of Ethernet trace (dotted line) with matched CMPP (solid line) (a) power spectrum (b) rate distribution (c) original filtered trace (d) CMPP sequence

4.3 Case Study

A comprehensive, blind case study of ISA was given in [6]. Here we just do an experimental study on the time growth of the ISA as a function of m at $N = 101$. At each given m, we take 100 randomly generated cases and measure the average CPU time per case. For a feasible case we measure the CPU time to find the first feasible solution; for an infeasible case we measure the total CPU time for complete search. The results are listed in Table 1. As m increases from 3 to 8, which is in the range of practical interest, the average CPU time grows from 0.027 minute to 5.10 minutes. Note that the possiblity to find a feasible solution for $m \geq 8$ at $N = 101$ becomes much reduced (less than 10%). For comparison purposes, the ratio of the CPU time at adjacent m is also provided in Table 1. After reaching the peak at $m = 5$, the growth rate of the average CPU time declines significantly as m further increases. The space complexity of the ISA is mainly measured by the memory requirement for the tableaus of the simplex. Each tableau takes $(n+1) \times (N+n-1)$ unites of data with double precision on level n. Approximately about $2n$ tableaus need to be stored in the memory, which amounts to less than 1 megabytes for $n \leq 8$ at $N = 101$. The space complexity is therefore not a problem for the ISA design.

5 Real Traffic Application

The proposed algorithms are applied to match the statistics of a wide spectrum of real traffic traces. We focus on the comparison of sequences generated from the matched CMPP with the original traces. The validity of the CMPP matching for queueing analysis was carefully examined in [9] by comparison of analytical queueing solutions of CMPP's to the simulated queuing solutions of original traces.

Since low-frequency traffic behavior (or, large time varying scales) has dominant impact on network performance as compared to high-frequency behavior [1], our statistical matching in

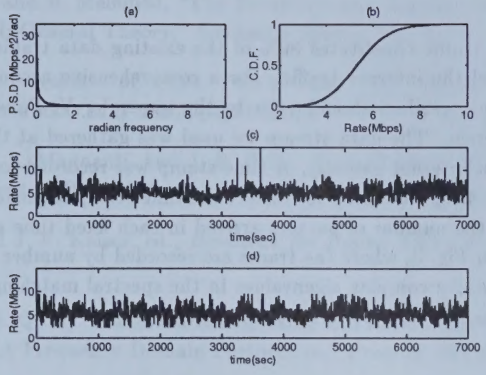


Figure 2: Comparison of a JPEG trace (dotted line) with matched CMPP (solid line) (a) power spectrum (b) rate distribution (c) original filtered trace (d) CMPP sequence

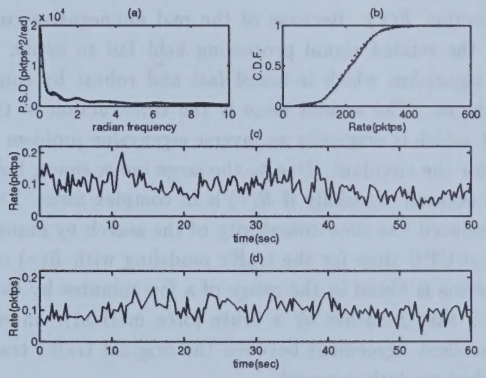


Figure 3: Comparison of a TCP trace (dotted line) with matched CMPP (solid line) (a) power spectrum (b) rate distribution (c) original filtered trace (d) CMPP sequence

this study are confined to the low-frequency range $\omega \in [0,\ 10]$ radian, which corresponds to the range of time scales longer than 0.6 second. In other words, all the traces are pre-filted by moving average operation for getting important low-frequency statistics.

We first take a 6-minute Ethernet data trace from the ftp site: thumper.bellcore.com. The results are displayed in Fig. 1. The dotted lines in Fig. 1a,b provide the collective power spectrum and rate cumulative distribution of the trace. The power spectrum, found by the proposed heuristic algorithm, consists of five distinctive complex eigenvalues ($m = 5$). But we cannot find a feasible CMPP for $N \leq 401$ unless with a certain degree of error tolerance as discussed in Section 4. Fig. 1 shows such a matched example by a CMPP at $N = 401$ with 30% error tolerance. While the matching error in power spectrum is relatively significant, its distribution matching is excellent due to the selection of a large N. The dynamics of the two traces in Fig. 1c,d are also found in a good agreement.

In the second example we take a 2-hour JPEG segment of the movie *Star wars* from the same ftp site at Bellcore. As one can see, the power spectrum highly concentrates on a low-frequency band, which is identified by four real eigenvalues plus one complex eigenvalue. A CMPP is matched at $N = 401$ with 0% error tolerance. The corresponding results are plotted

in Fig. 2.

Since the internet traffic constitutes 80% of the existing data traffic worldwide, it would be interesting to model the internet traffic. For a comprehensive account of the TCP traffic traces and its modeling, readers should refer to the paper by V. Paxson and S. Floyd [10] and the references therein. The data stream we used was gathered at the Lawrence Berkeley Laboratory's wide-area Internet gateway. A time stamp was recorded for each packet arrived. The data file is lbl-pkt-4.tcp. We need to convert the time stamp sequence into the packet rate sequence by counting the number of packets arrived in each fixed time slot (0.1 second). The results are presented in Fig. 3, where the traces are recorded by number of packets per second

There are four distinctive complex eigenvalues in the spectral matching. A feasible matched CMPP is found at $N = 301$ with 0% error tolerance.

6 Conclusion

This paper have developed algorithms to solve two most critical issues in measurement-based traffic modeling with the circulant structure. One is the parametric identification of the autocorrelation function $R(\tau)$. Because of the real nonnegative constraint $\psi \geq 0$, most existing techniques in the related signal processing field fail to apply. We have successfully developed a heuristic algorithm which is found fast and robust by comprehansive numerical study based on real traces. The second issue is the construction of traffic model to match with complicated $R(\tau)$, which is originally an inverse eigenvalue problem but transformed to an index search problem for the circulant. Due to the large index space, however, any brute force search method fails to succeed especially if $R(\tau)$ is in complex form. The proposed fast index search algorithm has reduced the time complexity of the search by many orders of magnitude. For example, the typical CPU time for the traffic modeling with $R(\tau)$ consisting of five or six complex exponential terms is found in the range of a few minutes by the proposed algorithms (which otherwise would take monthes by a brute force method). In real traffic application, our study shows an excellent agreement between the original traffic traces and the sequences generated by the matched analytical model.

Acknowledgement: One of the authors, H. Che, would like to thank Professor Patrick Jaillet for helpful discussion on design of the index search algorithm.

References

[1] S. Q. Li and C. Hwang, "Queue Response to Input Correlation Functions: Continuous Spectral Analysis," *IEEE/ACM Trans. Networking*, Vol. 1, No. 6, Dec. 1993, pp. 678-692.

[2] C. L. Hwang and S. Q. Li, "On the Convergence of Traffic Measurement and Queueing Analysis: A Statistical-MAtch Queueing (SMAQ) Tool," *IEEE Infocom'95 Conference*, April 1995, pp. 602-613.

[3] L. A. Kulkarni and S. Q. Li, "Measurement-Based Traffic Modeling: Capture Important Statistics," submitted to *Journal of Stochastic Model* (also available on web page:http://mocha.ece.utexas.edu/ sanqi/papers.html).

[4] A. Elwalid, D. Heyman, T. V. Lakshman, D. Mitra, and A. Weiss, "Foundamental Bounds and Approximations for ATM Multiplexers with Applications to Video Teleconferencing," *IEEE J. Select. Areas Commun.*, vol. 13, No.6, pp. 1004-1016,1995.

[5] D. L. Jagerman and B. Melamed, "The Transition and Autocorrelation Structure of TES Processes; Part I: General Theory," *Stochastic Models*, Vol. 8, No. 2, 1992, pp. 193-219.

[6] An extended version of this work can be found at the web-site: http://mocha.ece.utexas.edu/ sanqi/papers.html.

[7] R. K. Ahuja, T. L. Magnanti, and J. B. Orlin, *Network Flows: Theory, Algorithms, and Applications*, Prentice Hall, 1993.

[8] S. K. Mitra, and J. F. Kaiser, ed., *Handbook for Digital Signal Processing*,John Wiley & Sons, 1993.

[9] W. C. Lau and S. Q. Li, "Statistical Multiplexing and Buffer Sharing in Multimedia High-speed Networks:A Frequency Domain Perspective," *Proc. of GLOBECOM'95*, Dec. 1995.

[10] V. Paxson and S. Floyd, "Wide-area Traffic: The Failure of Poisson Modeling," *IEEE/ACM Transactions on Networking*, pp.226-244, June 1995.

Dynamic Channel Assignment For WDM Optical Networks With Little Or No Wavelength Conversion*

Ori Gerstel [1], Galen Sasaki [2], and Rajiv Ramaswami [1]

[1]IBM T.J. Watson Research Center
30 Saw Mill River Road, Hawthorne NY, 10532

[2]Department of Electrical Engineering, University of Hawaii
2540 Dole Street, Honolulu, HI 96822

Abstract

This paper proposes algorithms for dynamic assignment of wavelengths to connections (lightpaths) in optical wavelength division multiplexed networks. The worst-case situation is considered where no blocking is allowed, and there are no assumptions on the traffic arrival and holding times. The traffic is characterized by its *load* L, which is the maximum number of lightpaths that can be present on any link assuming no blocking.

Assignment algorithms are given that use at most $L \log_2 N$ wavelengths for rings and $2(L-1)\log_2 N$ for trees without wavelength conversion, where N is the number of network nodes. For rings, a simple *First-Fit* algorithm is shown to require at most $2.53L \log_2 N + 5L$ and at least $0.9L \log_2 N$ wavelengths. When limited wavelength conversion is allowed, there are arbitrary topology networks with a sufficient number of wavelengths that insure no blocking as long as the traffic load is kept to a constant fraction of the number of wavelengths, where the fraction is independent of N and L. In addition, we present a conversion pattern for rings with very limited conversion which requires $L \log_2 L + 4L$ wavelengths, thereby eliminating the dependence (in the no-conversion case) of the number of wavelengths on N. In a different traffic model where lightpaths are never taken down, the number of wavelengths needed is shown to be only $\max\{0, L-d\} + L$ for a conversion degree of d.

1 Introduction

In this paper we consider *wavelength division multiplexed* (WDM) optical networks. WDM is basically frequency division multiplexing in the optical frequency domain, where on a single optical fiber there are multiple communication channels. The channels are at different *wavelengths*, which correspond to carrier frequencies. These networks support *lightpaths*, which are end-to-end circuit-switched communication connections that traverse one or more links and use one WDM channel per link.

*This work was sponsored in part by grant MDA-972-95-C-0001 by DARPA.

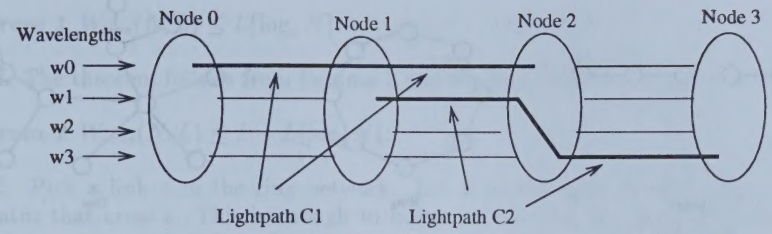


Figure 1: A WDM network with wavelengths $\{\omega_0, \omega_1, \omega_2, \omega_3\}$ and two lightpaths $C1$ and $C2$.

Figure 1 shows a WDM network that is composed of four nodes with optical fiber links, where each link has WDM channels at wavelengths $\{\omega_0, \omega_1, \omega_2, \omega_3\}$. Switching is done at each node so that channels may be connected to form lightpaths. Note that *wavelength conversion* devices are required to connect channels at different wavelengths. For example, lightpath $C2$ needs a wavelength converter at node 2, while lightpath $C1$ requires no wavelength conversion. The advantage of wavelength conversion is that WDM channels will be used more efficiently, but its disadvantage is increased cost and complexity.

In this paper we study the problem of allocating channels to lightpaths to insure no blocking. The network and traffic model assumes that lightpath requests arrive and terminate one at a time, and each request comes with its own *route*, i.e., the physical path that it should follow. Given the route, a sequence of free WDM channels, one per link, should be assigned to the lightpath. (Hence, existing lightpaths are undisturbed.) If such a sequence cannot be found then the request is *blocked*. This channel allocation problem is referred to as the *dynamic assignment* problem. The traffic is parameterized by its *load* L, which is defined to be the maximum number of lightpaths that can be on any link at any time assuming no blocking. Clearly, it is necessary that $L \leq W$, where W is the number of wavelengths, otherwise blocking will occur.

The model here is different from earlier approaches which allow blocking [4, 9, 10, 2, 8] and in some cases, assume that the traffic arrivals are Poisson [10, 2, 8]. The dynamic assignment problem in this context has been first studied in [5] for ring networks without wavelength conversion. [5] shows that any algorithm requires at least $\frac{L}{2}\log_2 N$ wavelengths to insure no blocking, and describes an algorithm that uses at most $3L\log_2 N$ wavelengths, where N is the number of nodes. In Section 2, we improve the upper bound to $L(\lceil \log_2 N \rceil + 1)$, and also consider tree networks. For tree networks without wavelength conversion, $2(L-1)\log_2 N$ wavelengths are sufficient to insure no blocking.

In Section 3, we consider a simple practical channel assignment algorithm called *First-Fit*, which has been shown to be efficient in simulation experiments [3, 4]. We show that First-Fit is good even in the worst case. In particular, we show that for First-Fit on a ring, approximately $2.53L\log_2 N + 5L$ wavelengths are sufficient to insure no blocking. Note that this is a tighter upper bound than the original upper bound given in [5] for large enough N. We also show that for ring networks First-Fit needs at least $0.9L\log_2 N$ wavelengths to insure no blocking.

In Section 4, we investigate how wavelength conversion can improve the utilization of channels. Our wavelength conversion model assumes that pairs of channels in adjacent links may be connected together to form lightpaths. We refer to a pair of channels that may be connected as being *compatible* since a communication signal on one may be switched and/or converted to the other. The *channel degree* of a network is the maximum number of compatible channels a channel may have on another link. The

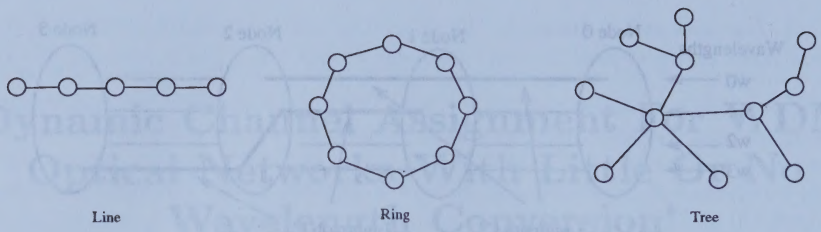


Figure 2: The line, ring, and tree topologies.

wavelength conversion capability of a network can be measured by its channel degree. For example, a network with full conversion capability has channel degree W, the number of wavelengths. While a network with no wavelength conversion has channel degree one. We show that if W is sufficiently large then there exists a conversion pattern between adjacent channels which enables networks with arbitrary topology to insure no blocking as long as the traffic load is at most δW, where $\delta > 0$ is some fraction independent of W and N. We also have results for a ring network with channel degree two. These results however, do not directly lead to practical solutions since δ is quite small.

In Section 4, we also consider the traffic model where lightpaths are never taken down. Note that for a ring network with no wavelength conversion, the number of wavelengths needed to insure no blocking is $3L-2$ [7], and there are examples which shows the number is necessary [6]. We extend results of [7] for a ring network with channel degree greater than one.

Conclusions are given in Section 5.

2 Dynamic Assignment For Rings And Trees With No Conversion

Three types of network topologies, shown in Figure 2, will be considered: the line, ring, and tree. The minimum number of wavelengths to insure no blocking for these three topologies will be given when there is no wavelength conversion.

Definition 1 *Let $W_{line}(N, L)$ (resp., $W_{ring}(N, L)$, and $W_{tree}(N, L)$) denote the number of wavelengths required to insure no blocking for any line (resp., ring, and tree) network with at most N nodes and no wavelength conversion, if the load across any link is at most L.*

Lemma 1 *If N is even then $W_{line}(N, L) \leq L + W_{line}(N/2, L)$.*

Proof. Note that there is a link e in the network whose removal leaves two line subnetworks X and Y, each with $N/2$ nodes. Let L wavelengths be dedicated to lightpaths that cross e. For these lightpaths, L wavelengths insures no blocking since there can be at most L of them at any time. Dedicate another $W_{line}(N/2, L)$ wavelengths to lightpaths that do not cross e, i.e., those that are entirely in X or Y. This insures no blocking since the subnetworks X and Y each have $N/2$ nodes, and the lightpaths in X can use the same wavelengths as the lightpaths in Y (because they do not intersect at any link). The total number of dedicated wavelengths to insure no blocking for all the lightpaths is $L + W_{line}(N/2, L)$. □

Theorem 1 $W_{line}(N, L) \leq L\lceil \log_2 N \rceil$.

Proof. The theorem follows from Lemma 1 and the fact that $W_{line}(1, L) = 0$. □

Theorem 2 $W_{ring}(N, L) \leq L + L\lceil \log_2 N \rceil$.

Proof. Pick a link e in the ring network. Let L wavelengths be dedicated to those lightpaths that cross e. This is enough to insure no blocking for these lightpaths since e can have at most L lightpaths. Dedicate another $L\lceil \log_2 N \rceil$ wavelengths to those lightpaths that do not cross e. These lightpaths can be viewed as ones that are in a line network with N nodes. Theorem 1 implies that these wavelengths are sufficient to insure no blocking. The total number of dedicated wavelengths to insure no blocking for all the lightpaths is $L + L\lceil \log_2 N \rceil$. □

Lemma 2 *For* $N \geq 1$, $W_{tree}(N, L) \leq 2L - 1 + W_{tree}(N/2, L)$.

Proof. Note that there exists a node v such that its removal leaves a collection of trees $T_1, T_2, ..., T_k$ (for some k) such that each tree has at most $N/2$ nodes [11]. Let $2L-1$ wavelengths be dedicated to those lightpaths that cross v, and let P_v denote those lightpaths that cross v. Note that each lightpath in P_v goes through exactly two links incident to v. Since there can be at most L lightpaths that can traverse any link, a lightpath in P_v can intersect with at most $2L - 2$ other lightpaths in P_v. Hence, $2L - 1$ wavelengths are enough to insure no blocking of lightpaths in P_v.

Now note that each lightpath that does not cross through v is entirely in one of the trees $T_1, T_2, ..., T_k$. Since lightpaths in different trees do not intersect, they may use the same wavelengths. Since each tree has at most $N/2$ nodes, $W_{tree}(N/2, L)$ wavelengths suffices to insure no blocking for lightpaths that do not cross through v.

The total number of wavelengths for all lightpaths to insure no blocking is at most $2L - 1 + W_{tree}(N/2, L)$. □

Theorem 3 $W_{tree}(N, L) \leq (2L - 1)\lceil \log_2 N \rceil$.

Proof. The theorem is implied by Lemma 2 and the fact that $W_{tree}(1, L) = 0$. □

3 First-Fit Assignment On A Ring With No Conversion

First-Fit is an algorithm for assigning a wavelength to a lightpath in a network with no wavelength conversion. It assumes that the wavelengths are labeled $0, 1, ...W - 1$, and assigns to a lightpath a wavelength with the lowest label that is available in each link of the lightpath. Upper and lower bounds on the worst case number of wavelength to insure no blocking for First Fit on a ring will be presented.

3.1 Upper Bound

Definition 2 *Let* $W^{FF}_{ring}(H, L)$ *denote the maximum label of a wavelength used by a lightpath of length at most* H *on any ring network with load at most* L *and that uses First-Fit.*

Lemma 3 *For* $H \geq 1$, $W^{FF}_{ring}(3H, L) \leq 4L - 3 + W^{FF}_{ring}(H, L)$.

Proof. It will be shown that a lightpath p of length $k \leq 3H$ will be assigned a wavelength whose label is at most $4L - 3 + W^{FF}_{ring}(H, L)$. Consider the case $k \leq H$. Then the lightpath can be assigned to a wavelength at most $W^{FF}_{ring}(H, L)$, which in turn is at most $4L - 3 + W^{FF}_{ring}(H, L)$.

Now consider the case $H < k \leq 3H$. It will be shown that there is a wavelength in the set $W^* = \{W^{FF}_{ring}(H, L) + 1, W^{FF}_{ring}(H, L) + 2, ..., W^{FF}_{ring}(H, L) + 4L - 3\}$ that can be assigned to p. Let $p^* = \{p_1, p_2, ..., p_m\}$ be the set of lightpaths already in the network that intersect p and use wavelengths from the set W^*. Since these lightpaths use wavelengths greater than $W^{FF}_{ring}(H, L)$, their lengths must be greater than H.

Let $e_1, e_2, ..., e_k$ be the sequence of links of the lightpath p. Let $e_1, e_{\lceil k/3 \rceil}, e_{\lceil 2k/3 \rceil}, e_k$ be referred to as the *critical* links for p. Note that each lightpath of p^* has length greater than H, which in turn is at least $k/3$. Thus, each lightpath of p^* intersects at least one of the critical links. Let $n_1, n_{\lceil k/3 \rceil}, n_{\lceil 2k/3 \rceil}$, and n_k be the number of lightpaths of p^* that cross $e_1, e_{\lceil k/3 \rceil}, e_{\lceil 2k/3 \rceil}$, and e_k, respectively. Since each lightpath of p^* intersects at least one of the critical links, $|p^*| \leq n_1 + n_{\lceil k/3 \rceil} + n_{\lceil 2k/3 \rceil} + n_k \leq 4(L - 1)$. Since the number of wavelengths of W^* is $4L - 3$, there must be an available wavelength in W^* for p. Thus, p is assigned a wavelength labeled at most $4L - 3 + W^{FF}_{ring}(H, L)$. □

Theorem 4 $W^{FF}_{ring}(H, L) \leq (4L - 3)\lceil \log_3 H \rceil + L$.

Proof. The theorem follows from Lemma 3, and the fact that $W^{FF}_{ring}(1, L) = L$. □

Corollary 1 *Consider a ring network with N nodes and using the First-Fit algorithm. Then $2.53 \cdot L \log_2 N + 5L$ wavelengths insures no blocking if the load across any link is at most L.*

3.2 Lower Bound

In [5] it was proven that any algorithm that assigns wavelengths to lightpaths in an optical ring with N nodes and no wavelength conversion, requires $W \geq 0.5L \log_2 N$ wavelengths in the worst case. We now prove that if the algorithm is assumed to be First-Fit, a better lower bound on the worst case performance can be achieved for reasonable numbers of nodes: $W \geq 0.9L \log_2 N$.

Consider the following pattern of lightpath add/delete requests, depicted in Figure 3:

Phase 1. A single hop lightpath addition request to connect nodes 0 and 1 arrives and is allocated wavelength 0.

Phase 2. A single hop request to connect nodes 1 and 2 arrives and is allocated wavelength 0 by First-Fit. Another such request arrives and is allocated wavelength 1. Next, the first lightpath between nodes 1 and 2 is deleted. The current configuration is two single hop lightpaths with a load of 1 using two wavelengths.

Phase 3. A two hop lightpath from node 0 to node 2 is requested, and is allocated wavelength 2. Next, the one hop lightpaths are deleted and Phases 1–2 are repeated between nodes 2 and 4.

Phase i. After phase $i - 1$, wavelengths $\{0, 1, ..., i - 2\}$ have been used. In order to establish a short lightpaths that uses wavelength i it is necessary to rearrange the relative order of lightpaths in the configuration of Phase $i - 1$ (the details of the

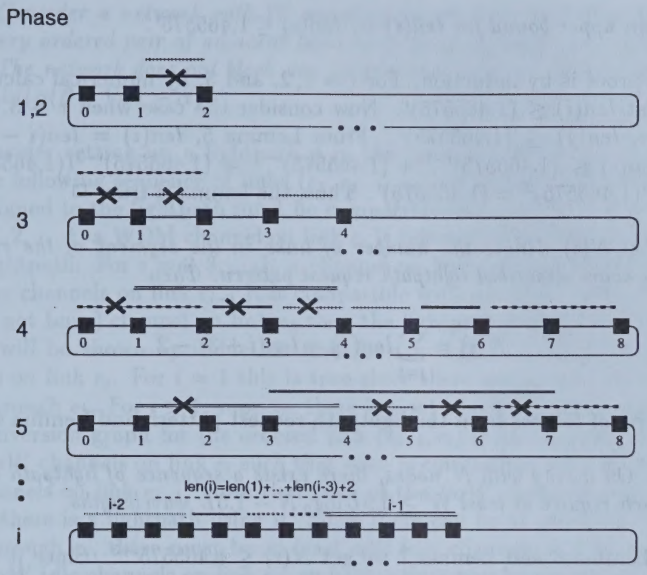


Figure 3: Worst case scenario for First-Fit

altered schedule of requests is left to the reader). In the modified configuration, the lightpath which uses wavelength $i-2$ is the leftmost one and the lightpath that uses $i-1$ is the rightmost one. The new request overlaps one link of the leftmost and rightmost lightpaths (and all the intermediate lightpaths). Thus First-Fit allocates wavelength $i-1$ to it.

Lemma 4 *Let $len(i)$ denote the length of the lightpath added at Phase i (the one that uses the i^{th} wavelength). Then $len(1) = 1, len(2) = 1, len(3) = 2$, and for $i > 3$,*

$$len(i) = 2 + \sum_{j=1}^{i-3} len(j) \quad .$$

Proof. For $i = 1, 2$, and 3, the lemma is easy to check. For $i > 3$, the lightpath of the lemma spans all the lightpaths from phases 1 to $i-3$, which are adjacent and non-overlapping. In addition it shares one link with the lightpath from phase $i-2$ and one link from the lightpath of phase $i-1$. □

It is interesting to note the following similarity between $len(i)$ and Fibonacci series:

Lemma 5

$$len(i) = \begin{cases} 1, & \text{if } i = 1, 2, \\ 2, & \text{if } i = 3, \\ len(i-1) + len(i-3), & \text{otherwise.} \end{cases}$$

Proof. It follows from Lemma 4 and, for $i > 3$,

$$len(i) - len(i-1) = (2 + \sum_{j=1}^{i-3} len(j)) - (2 + \sum_{j=1}^{i-4} len(j)) = len(i-3) \quad .$$

□

Lemma 6 *An upper bound for len(i) is:* $len(i) \leq 1.465575^i$.

Proof. The proof is by induction. For $i = 1, 2$, and 3, by numerical calculation it can be shown that $len(i) \leq (1.465575)^i$. Now consider the case when $i > 3$, and suppose for all $j < i$, $len(j) \leq (1.465575)^j$. From Lemma 5, $len(i) = len(i-1) + len(i-3)$. Thus, $len(i) \leq (1.465575)^{i-1} + (1.465575)^{i-3} = (1.465575)^{i-3}((1.465575)^2 + 1) \leq (1.465575)^{i-3}(1.465575)^3 = (1.465575)^i$. Thus, the lemma is true. □

Lemma 7 *Let $N(i)$ denote the number of links in the segment of the ring used by i phases of the above described lightpath request pattern. Then*

$$N(i) = \sum_{j=1}^{i} len(j) = len(i+3) - 2 \quad .$$

Proof. The proof follows from the lightpath request pattern and Lemma 4. □

Theorem 5 *On a ring with N nodes, there exists a sequence of lightpath additions and deletions which require at least* $W \geq 0.9L \log_2 N - 1.5L$ *wavelengths.*

Proof. By Lemma 6 and Lemma 7 we get $N(i) \leq 1.465575^{i+3}$. Since it is possible to apply Phase i only if $N \geq N(i)$ we get $\log_{1.465575} N \geq i + 3$ or $i \leq \log_{1.465575} N - 3 = \frac{\log_2 N}{\log_2 1.465575} - 3 < 1.81335 \log_2 N - 3$. Since the above scenario had $\lambda_{\max} = 2$, it is possible to duplicate it $L/2$ times and require $iL/2$ wavelengths. Thus the number of necessary wavelengths is $W \geq 1.81335/2L \log_2 N - 1.5L$. □

4 Limited Conversion

In this section we determine the performance of networks with limited wavelength conversion, i.e., the channel degree of the network is small. We will first consider networks with arbitrary topology, and then networks with the ring topology.

We will use the following terminology. Consider a network with wavelengths {0, 1, ..., $W-1$}, and channels are numbered according to their wavelengths. For an ordered pair of adjacent links x and y in the network, a bipartite graph $G_{x,y} = (V_x, V_y, E)$ is called its *conversion graph* if $V_x = V_y = \{0, 1, ..., W-1\}$ and for each $(i, j) \in E$, channel i on link x is compatible with channel j on link y. (Note that for a network with full wavelength conversion, the conversion graphs are complete bipartite graphs, while for a network with no wavelength conversion, the conversion graphs have edges $E = \{(i, i) | 0 \leq i < W\}$.) A conversion graph is said to be *symmetric* if $(i, j) \in E$ implies $(j, i) \in E$.

Our performance result (Theorem 6) depends on conversion graphs with particular *expansion* properties, stated next.

Definition 3 *Consider a bipartite graph (V_1, V_2, E) with each node having at most d incident edges. For each subset of nodes $S \subseteq V_1$, let $\Gamma(S)$ denote the subset of nodes in V_2 that are adjacent to a node in S (i.e., $\Gamma(S) = \{j \in V_2 : i \in V_1, (i, j) \in E\}$). The graph is called an (α, β, d)-expander, for some $0 < \alpha < \frac{1}{2}$ and $\beta > 1$, if for each subset of nodes $S \subseteq V_1$ such that $|S| \leq \alpha |V_1|$, $|\Gamma(S)| \geq \beta |S|$.*

Lemma 8 *[1] There is a triple (α, β, d), where $0 < \alpha < \frac{1}{2}$ and $\beta > 1$, such that for each n that is sufficiently large, there is a symmetric (α, β, d)-expander with n nodes.*

Lemma 9 *Consider a network with W wavelengths per link, such that the conversion graph for every ordered pair of adjacent links is an (α, β, d)-expander, where $0 < \alpha < \frac{1}{2}$ and $\beta > 1$. The network does not block any lightpaths as long as the load is at most δW, where $\delta = \min\{\alpha(\beta - 1), 1 - \alpha\}$.*

Proof. Consider setting up a lightpath p in the network, and suppose the lightpath traverses the following sequence of links $(e_1, e_2, ..., e_k)$ for some k. Note that the WDM channels assigned to the lightpath must be compatible from link to link.

For $i = 1, 2, ..., k$, a WDM channel on link e_i is referred to as being *busy* if it is being used by a lightpath. For $i = 2, 3, ..., k$, a channel on link e_i is also referred to as being *busy* if all the channels on link e_{i-1} it is compatible with are busy. Note that if there is an *idle* (i.e., not busy) channel on link e_k then the lightpath p may be set up.

Next, it will be shown by induction that for $i = 1, 2, ..., k$, there are at least αW idle channels on link e_i. For $i = 1$ this is true since there are at most $\delta W \leq (1 - \alpha)W$ lightpaths through e_1. For $i > 1$, suppose there is a set of αW idle channels on link e_{i-1}. Since the conversion graph for the ordered pair (e_{i-1}, e_i) is an (α, β, d)-expander, there is a set of $\beta\alpha W$ channels on link e_i such that each is compatible with at least one of the αW idle channels on link e_{i-1}. Note that each of the $\beta\alpha W$ channels on link e_i may be busy only if there is a lightpath using it. Since there can be at most $\delta W \leq (\beta - 1)\alpha W$ lightpaths through e_i, there must be at least αW idle channels in link e_i. Thus, there are at least αW idle channels on link e_k, so lightpath p may be set up. □

Theorem 6 *There is a fraction $\delta > 0$ and integer $d > 0$ such that for any network with sufficiently large number of wavelengths W, the network can have channel degree d and insure no blocking of lightpaths if the traffic load is at most δW.*

Proof. The theorem follows directly from Lemmas 8 and 9. □

For the rest of this section, we focus our attention on ring networks. The next Theorem 7 is for a ring network with channel degree 2. The subsequent Theorem 8 is for a ring network with channel degree $d > 1$ and where lightpaths are set up but never taken down.

Theorem 7 *Let L and N be arbitrary positive integers. Let $W = L\lceil \log_2 L \rceil + 4L$. There is a ring network with N nodes, W wavelengths, and channel degree 2 that that does not block any lightpaths as long as the load is at most L.*

The outline of the proof of the theorem is left in Appendix A.

Theorem 8 *Consider a traffic model where lightpaths may be set up but never taken down. Let L and d be integers that satisfy $L \geq 1$ and $d > 1$. Then there is a ring network with channel degree d and $\max\{0, L - d\} + L$ wavelengths that does not block any lightpaths as long as the load is at most L.*

The theorem can be proven in a straightforward way by modifying the results in [7]. For completeness, we provide an outline of the modifications in the Appendix B.

5 Conclusions

The worst-case model used in this paper determines the maximum traffic load that can be supported without any blocking. We showed that the simple first-fit algorithm, which

does well in simulations that allow blocking, is also quite good in the worst-case, requiring at most $2.53L \log_2 N + 5L$ wavelengths in a ring network without wavelength conversion. A better algorithm for this case uses only $L \log_2 N$ wavelengths.

We also showed that limited wavelength conversion can increase the utilization of WDM channels. Our results show that the number of wavelengths for no blocking is dependent on the traffic load, and not dependent on the number of nodes. This is better than the numbers of wavelengths for trees and rings with no wavelength conversion that are functions of $\log_2 N$. Thus, limited wavelength conversion enables more scalable solutions.

For the case when lightpaths are never taken down, very limited conversion of channel degree 2 enables to achieve 33% decrease in the number of wavelengths that are needed to support a given load, and this number linearly decreases (at least) with increased channel degree.

A Outline Of The Proof Of Theorem 7

The following is an outline of the proof of Theorem 7. To prove the theorem we will define a ring network with N nodes, $L\lceil \log_2 L \rceil + 4L$ wavelengths, and channel degree 2 that will insure no blocking as long as the maximum load is at most L. We may assume that $N \geq L$, otherwise, Theorem 2 implies Theorem 7.

The ring has nodes numbered $0, 1, ..., N-1$ going clockwise, and for $i = 0, 1, ..., N-1$, the link between nodes i and $(i+1) \bmod N$ is numbered i. The ring is partitioned into segments, each having at least L links but less than $2L$ links. (Note that such a partitioning is always possible.) Now $L\lceil \log_2 L \rceil + L$ wavelengths are dedicated to lightpaths that do not cross segments, and these wavelengths do not have wavelength conversion. Since the lightpaths are confined to segments with less $2L$ links, Theorem 1 implies that there will be no blocking of these lightpaths.

The other $3L$ wavelengths are dedicated to supporting lightpaths that cross segments, i.e., *inter-segment* lightpaths. These wavelengths have wavelength conversion. To describe how the WDM channels are compatible, a graph G^* will be used, in which WDM channels are compatible if they are incident to a common vertex. It is straight forward to check that this ring network has channel degree two.

- **The vertices of G^*:** There are N stages of vertices where each stage has $2L$ vertices, and stage i represents node i in the ring network. In each stage i, there are two types of vertices called *u-vertices* and *v-vertices* and are labeled $\{u_0(i), u_1(i), ..., u_{L-1}(i)\}$ and $\{v_0(i), v_1(i), ..., v_{L-1}(i)\}$ respectively.

- **The edges of G^*:** For $i = 0, 1, ..., N-1$, there are $3L$ edges between stage i and stage $(i+1) \bmod N$ vertices corresponding to the $3L$ WDM channels on link i. The $3L$ channels are of three types: *shift channels*, *u-channels*, and *v-channels*, where there are L of each type. Figure 4 shows a subgraph of G^* corresponding to a segment of the ring network with links $m, m+1, ..., n-1$. For $i = m, m+1, ..., n-1$, the channels between the stage i and stage $i+1$ vertices (i.e., the channels of link i) are as follows:

  - Each u-vertex $u_j(i)$ (for $0 \leq j < L$) has a u-channel between it and $u_j(i+1)$. It also has a shift channel between it and

$$\begin{cases} u_{(j+1) \bmod L}(i+1), & \text{if } i < n-1 \\ v_j(i+1), & \text{if } i = n-1 \end{cases}$$

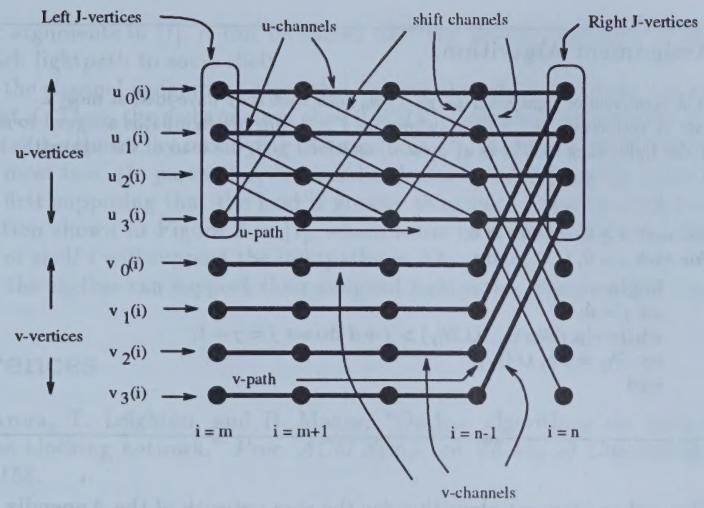


Figure 4: The subgraph of G^* corresponding to the segment of the ring with links $\{m, m+1, ..., n-1\}$, and $L = 4$. The u-channels and v-channels are shown as thick lines, while the shift channels are shown as thin lines.

– Each v-vertex $v_j(i)$ (for $0 \leq j < L$) has a v-channel between it and

$$\begin{cases} v_j(i+1), & \text{if } i < n-1 \\ u_j(i+1), & \text{if } i = n-1 \end{cases}$$

Next, we describe how inter-segment lightpaths are assigned channels, and begin by introducing some terminology. Consider a segment of the ring network and its corresponding subgraph of G^* as shown in Figure 4. First, the u-vertices in stages m and n will be referred as *joining vertices* or *J-vertices* because they "join" segments together. The J-vertices in stages m and n are called the *left* and *right* J-vertices, respectively. A J-vertex is *busy* if it has a lightpath going through it, and is *idle* otherwise. Second, note that u-channels form L paths of channels crossing the segment, and these will be called *u-paths*. Similarly, the v-channels form L paths, and these paths will be referred to as *v-paths*. A u-path or v-path is *busy* if there is a lightpath on it, and *idle* otherwise. An idle u-path or v-path is called *available* if the right J-vertex that it is connected to is also idle.

Inter-segment lightpaths are assigned to channels as follows. The assignment for a lightpath is done segment by segment, starting from one end of the lightpath and going clockwise around the ring network. In the first segment, the lightpath follows an available v-path to the corresponding idle (right) J-vertex. In an intermediate segment, the lightpath starts from an idle (left) J-vertex, follows shift channels until it reaches an available u-path, and then follows the u-path to an idle (right) J-vertex. In the final segment, the lightpath starts from an idle (left) J-vertex and follows shift channels.

We now argue that the assignment works, i.e., a lightpath will not be assigned channels already used by existing lightpaths. First note that in a segment, if a lightpath follows a u-path or v-path then it goes through a right J-vertex. Thus, the existence of an idle (right) J-vertex implies there is an available u-path and v-path. Since the load is L and there are L right J-vertices, if a lightpath is to be set up through a right J-vertex then

Channel Assignment Algorithm:

- **Input:** A sequence of lightpaths $p_0, p_1, ..., p_n$ such that they have load at most L.
- **Output:** A collection $Sh_0, Sh_1, ...$, where for $i \geq 0$, Sh_i are lightpaths assigned to shelf i. The load of the lightpaths in Sh_0 is at most d, and for $i > 0$, the load of the lightpaths in Sh_i is at most 2.
- **begin**

for each $i \geq 0$, set $Sh_i = \emptyset$;
for each $i = 0, 1, ..., n$, **do**
begin
set $j = 0$;
while $r(p_i / Sh_0 \cup ... \cup Sh_j) > j + d$ **do** set $j = j + 1$;
set $Sh_j = Sh_j \cup \{p_i\}$;
end

end

Figure 5: Channel assignment algorithm for the ring network of the Appendix and Theorem 7.

there is at least one idle right J-vertex. Therefore, there is at least one available u-path and v-path.

Now consider a lightpath p that is about to be assigned channels, and consider the segments it traverses. In its first segment, an available v-path can be found for p. In an intermediate segment, p starts from an idle left J-vertex, and can proceed along shift channels without overlapping an existing lightpath. This is possible since all the existing lightpaths that start from the left J-vertices, first follow the shift channels before going along a u-path. Since the segment has at least $L - 1$ links, the lightpath p can reach all u-paths by just following shift channels. In particular, it will be able to reach an available u-path. In the final segment, p starts from an idle left J-vertex, and so it can follow a sequence of shift channels without overlapping an existing lightpath. Therefore, the lightpath will not overlap with an existing one.

B Outline Of The Proof Of Theorem 8

We will discuss how the results of [7] can be modified to prove Theorem 8. The ring network in the theorem has W WDM channels per link, where $W = \max\{0, L - d\} + L$ and L is the load of the lightpaths. The channels are numbered $\{0, 1, ..., W - 1\}$. The channels will be partitioned into sets called *shelves*. If $d \geq L$ then there is only one shelf and it is numbered 0. If $d < L$ then there are $L - d + 1$ shelves and they are numbered $0, 1, ..., L - d$. Shelf 0 consists of the channels numbered $\{0, 1, ..., d - 1\}$, and for $i = 1, 2, ..., L-d$, shelf i consists of the pair of channels $\{d+2(i-1), d+2(i-1)+1\}$. In the ring network, channels that are in the same shelf and in adjacent links are compatible with one another.

The channel assignment algorithm for the network is shown in Figure 5, and is a modification of the algorithm *COLOR* in [7]. The algorithm assigns incoming lightpaths to shelves. In the algorithm, Sh_i denotes the collection of lightpaths that have been assigned to shelf i. Also, for a lightpath p, $r(p/Sh_0 \cup ... \cup Sh_i)$ denotes the value $\max_e r(e/Sh_0 \cup ... \cup Sh_i)$, where the maximum is taken over all links e traversed by p, and $r(e/Sh_0 \cup ... \cup Sh_i)$ denotes the number of lightpaths in $Sh_0 \cup ... \cup Sh_i$ that traverse

e. Using arguments in [7], it can be shown that the channel assignment algorithm will assign each lightpath to some shelf.

Note the channel assignment algorithm insures that the load of the lightpaths in Sh_0 be at most d. Thus, the d channels of shelf 0 will support the lightpaths in Sh_0. Also, for $i > 0$, it can be shown, by modifying the results in [7], that the load for the lightpaths in Sh_i is at most two. (In particular, this can be shown by modifying the proof of Lemma 4 in [7] by first supposing that the load is greater than two. Then Lemma 3 in [7] implies the situation shown in Figure 2 in [7], which leads to a contradiction.) Thus, the two channels of shelf i will support the lightpaths in Sh_i.

Since the shelves can support their assigned lightpaths, Theorem 8 is implied.

References

[1] S. Arora, T. Leighton, and B. Maggs, "On-line algorithms for path selection in a non-blocking network," *Proc. ACM Symp. on Theory of Computing* (1990), pp. 149-158.

[2] A. Birman, "Computing Approximate Blocking Probabilities for a Class of Optical Networks," *IEEE J. Selected Areas in Communications,* vol. 14, no. 5, pp. 852–857, June 1996 (An earlier version appeared in Proc. Infocom '95).

[3] A. Birman and A. Kershenbaum, "Routing and wavelength assignment methods in single-hop all-optical networks with blocking," *Proc. IEEE Infocom 95*, pp. 431-438, 1995.

[4] R. Barry and P. Humblet, "Models of blocking probability in all-optical networks with and without wavelength changers," *IEEE J. Selected Areas in Communications*, vol. 14, no. 5, pp. 858-867, June 1996.

[5] O. Gerstel and S. Kutten, "Dynamic wavelength allocation in all-optical ring networks," Research Report, IBM Research Division, Watson Research Center, RC 20462, May 1996. Also in `http://www.research.ibm.com/wdm/route/`.

[6] H. Kierstead and W. Trotter, "An extremal problem in recursive combinatorics," *Congressus Numerantium*, vol. 33, pp. 143-153, 1981.

[7] M. Slusarek, "Optimal online coloring of circular arc graphs," *Informatique Theoretique et Applications*, vol. 29, no. 5, pp. 423-429, 1995.

[8] M. Kovacevic and A. S. Acampora, "On the benefits of wavelength translation in all optical clear-channel networks," *IEEE J. Lightwave Technology*, vol. 14, no. 6, pp. 1264-1277, June 1996 (An earlier version appeared in Proc. Infocom'95).

[9] R. Ramaswami and K. N. Sivarajan, "Routing and wavelength assignment in all-optical networks," *IEEE/ACM Trans. on Networking,* Oct., 1995, pp. 489–500 (An earlier version appeared in Infocom'94).

[10] J. Yates and J. Lacey and D. Everitt and M. Summerfield, "Limited-range wavelength translation in all-optical networks," *Proc. Infocom 96*, pp. 954–961, 1996.

[11] B. Zelinka, "Medians and peripherians on trees," *Arch. Math. (Brno)*, pp. 87-95, 1969.

Algorithms for Establishing and Maintaining Sessions in Wireless Networks with Changing Connectivity *

Anastassios Michail and Anthony Ephremides
Electrical Engineering Department
and Institute for Systems Research
University of Maryland
College Park, MD 20742 USA
tassos,tony@eng.umd.edu

October 1, 1996

Abstract

In future wireless networks in which all nodes are mobile, and hence subject to variable connectivity, it is necessary to develop algorithms for identifying and maintaining paths between communicating pairs of nodes. Based on earlier work that accomplishes such a task for datagram packet service, we develop similar algorithms for connection oriented service. The algorithms are distributed and rely on limited exchange of messages. In addition, in order to maintain on-going sessions in the presence of topological changes, and to avoid the use of power control (which may aggravate interference and cause more connectivity fluctuations), we opt for a method of "softening" these changes by means of rate control. In this paper we describe these algorithms which are presently undergoing extensive simulation-based evaluation.

1 Introduction

We consider the routing problem in all-mobile wireless networks which are subject to frequent and unpredictable connectivity changes. We are interested in establishing and maintaining paths between mobile nodes for both connection-less datagram service and connection-oriented service. In all-mobile wireless networks the status of the communication links is a function of nodes' location, distance, transmission power levels, channel interference and antenna patterns. Without assuming any known network architecture (e.g. cellular) we propose a distributed algorithm that is designed to perform for any set of nodes with the same functionalities and properties without the need of global connectivity information.

*Prepared through collaborative participation in the Advanced Telecommunications/Information Distribution Research Program (ATIRP) Consortium sponsored by the U.S. Army Research Laboratory under Cooperative Agreement DAAL01-96-2-0002.

Our approach is based on a distributed routing algorithm for data that was presented in [1] and to which we will refer as CE algorithm. This algorithm assumes adequate bandwidth and an underlying access protocol that is interference-free. The algorithm executes in two logical phases which occur simultaneously in a dynamic topology. In the "construction phase" a source node generates a "query" message that is flooded throughout the network in search of a path to a specific destination node. A "reply" message is generated by the destination node, or any other node having a route to the destination, and is flooded in a reverse manner. The query-reply propagation results in an initial set of routes to the specific destination. In the "maintenance phase" the algorithm reacts in a similar way to link failures by the exchange of control packets in order to erase invalid routes and quickly construct new sets of routes. The algorithm is deadlock-free and capable of detecting catastrophic separation of the network.

In this paper we propose an algorithm that identifies and maintains paths between communicating pairs of nodes that desire connection oriented service. The algorithm establishes sessions between mobile nodes using paths among those provided by the underlying CE algorithm which satisfy the bandwidth requirements, and converts connectivity changes to "soft" failures by utilizing a transmission rate adjustment mechanism at the receiver which maintains on-going sessions at the cost of a lower transmission rate, and allows time for the discovery of a new path. The algorithm must avoid loops and deadlocks as in [1] and, in this case, it is modified to include in the message exchange information on the availability of bandwidth at each node during the establishment of a route.

The rest of the paper is arranged as follows. Section 2 gives a high-level description of the algorithm followed by a detailed presentation of its operation, rules, and features. In section 3 we briefly present the simulation setup and performance measures. Finally, section 4 discusses the algorithm and possible future modifications and enhancements.

2 Algorithm description

2.1 Overview of operation

We consider multiple independently executing versions of the distributed routing CE algorithm, each one of them running for a specific destination. A corresponding version of the new algorithm is overlaid on top of each such version and utilizes information provided by the underlying routing protocol to request connections through existing routes. We describe the algorithm for one such version that runs for a destination node denoted DEST.

An underlying link-level protocol is assumed which assures distributed knowledge of the connectivities' changes. The mechanism of this link-level protocol operation will be addressed at a future time. At this point we assume that each node i is aware of all its adjacent nodes at all times. Additionally, transmitted packets are received correctly and simultaneous two-way transmission over a link that would cause interference does not occur. Each node is characterized by a unique node identifier. All the nodes adjacent to a node i are referred to as its neighbors. For each neighbor j of node i there exists a link between j and i which can either be undirected or directed. In the second case, if the link is directed from i to j it is characterized as downstream (DN) and j is a downstream neighbor of i. Similarly, if the link is directed from j to i it is characterized as upstream (UP) and j is an upstream neighbor of i.

Our algorithm's execution can be viewed in two logical phases which occur simultaneously in a dynamic topology, the "Construction phase" and the "Maintenance phase". During those phases, the algorithm uses information for the existing paths to the DEST to establish (construction phase) and maintain (maintenance phase) sessions according to availability of network resources.

In wireless networks bandwidth constraints are determined by the availability or the lack of transceivers, in contrast to fixed-link networks where bandwidth availability depends on link capacity. Mobile node i can transmit information to a neighboring mobile node j only if j has at least one transceiver in an idle state. Hence the existence of a route from a source node to the DEST is not sufficient to assure a successful connection request. We propose a technique which takes into account these restrictions in bandwidth, to gradually build and update a database at each node that supplies information about availability of resources throughout one or multiple paths initiating from the same node to the DEST.

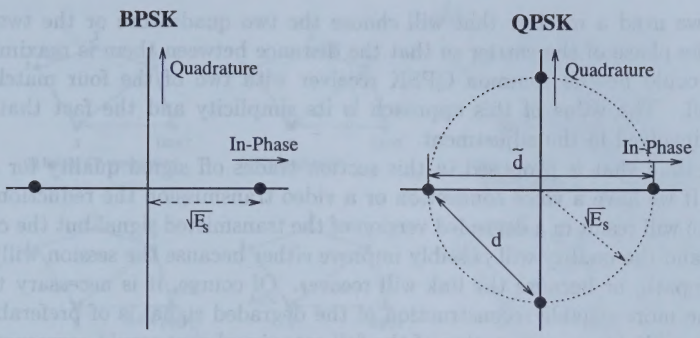
2.1.1 Construction Phase

During the construction phase connection requests are placed individually by mobile nodes in the network and sessions are established, using among the existing paths those that have the required number of available transceivers. At the beginning no bandwidth information is available at the nodes and the selection of the "next-hop" node in the process of exploring the routes is made randomly until a path from the transmitting source to the DEST is found that can accommodate the connection request. The acknowledgments that are generated by the DEST to inform the source that the session can be initiated are also flooded in the network to inform other nodes that have routes to the DEST about the maximum number of transceivers available in those paths.

Each node compares the maximum number of available transceivers, in the path initiating from its location to the DEST, with the number of its idle – at that moment – transceivers, and broadcasts to its upstream nodes the minimum value. That way a database is built and updated at each node that keeps track of the availability of resources in every route to the DEST. The selection of the "next-hop" node in the route exploration phase during the construction phase can thus be based on the current information of the node which increases the probability of a success in the search for a path that can accommodate the connection request.

2.1.2 Maintenance Phase

During the maintenance phase the algorithm reacts to connectivity changes that affect on-going sessions. Because a voice or video connection must not be "dropped" or interrupted the algorithm takes advantage of the possibility to convert a wireless connectivity change to a "soft" failure, and maintains the on-going connection until a new path is discovered. When a "hard failure" takes place, connectivity is completely lost and nothing can be done to maintain the connection. However, in a wireless environment a failure usually occurs when the energy-per-bit-to-interference plus noise ratio $\frac{E_b}{N_o}$ falls below a predetermined threshold. In order to convert a failure to a soft failure we can either utilize a power control mechanism or adjust the transmission rate. Power control may aggravate interference and cause more connectivity fluctuations. Thus we propose to adjust information rate in such a way that given transmission power and interference plus noise we maintain $\frac{E_b}{N_o}$ above the predetermined threshold.



(a) 2-phase signal constellation (b) 4-phase signal constellation

Figure 1: BPSK and QPSK constellation diagrams

As a result of this mechanism a session can be maintained at the cost of a lower transmission rate while the algorithm searches for alternative paths to re-route the connection. Thus we temporarily trade off signal quality for algorithm efficiency, which is possibly acceptable when the application is a voice connection or a video transmission that can temporarily afford loss of bits –which implies quality degradation– instead of being interrupted. Of course, we cannot apply this technique when data files are being transmitted.

In the subsequent section we discuss the details of the transmission rate adjustment mechanism that we propose. In the detailed description of the algorithm we explain how a node reacts when a new path is required to re-route an on-going connection.

2.2 Transmission rate adjustment mechanism

The problem of the rate adjustment is approached by a technique based on the digital modulation scheme that is used and on the receiver characteristics. We discuss this mechanism for a Phase Shift Keying (PSK) modulation scheme and for purposes of illustration we assume that in our network the transmitted information is encoded in Quadrature Phase Shift Keying (QPSK). As we mentioned above we need a mechanism that will adjust transmission rate so that the ratio $\frac{E_b}{N_o}$ is maintained above a predetermined threshold. Instead of a node –which senses a quality degradation in one of its DN links– having to notify the transmitting source to adjust the rate by exchange of control packets, it starts decoding the received signal as a Binary PSK. The phase of the carrier takes on 1 of 4 equally spaced values as we can see on the constellation diagram in figure 1. Each value of phase corresponds to a unique pair of message bits. In BPSK the phase of the carrier can take on 1 of 2 values, each one of them corresponding to one message bit. The distance between two adjacent signals is $d = \sqrt{2E_s}$ while the distance between two signals having a phase difference of 180 degrees is $\tilde{d} = 2\sqrt{E_s}$, where E_s is the energy per symbol. If the ratio $\frac{d}{N_o}$, where N_o represents interference plus noise, falls below a predetermined threshold T, the link is considered to have a low quality and the rate adjustment mechanism is activated. So, d is automatically replaced by $\tilde{d}$ which could raise the ratio $\frac{\tilde{d}}{N_o}$ above the threshold.

Hence we need a receiver that will choose the two quadrature or the two in-phase values of the phase of the carrier so that the distance between them is maximum. Such a receiver could be the common QPSK receiver with two of the four matched filters switched off. The value of this approach is its simplicity and the fact that only the receiver is involved in the adjustment.

The method that is proposed in this section trades off signal quality for algorithm efficiency. If we have a voice connection or a video transmission the reduction in transmission rate will result in a degraded version of the transmitted signal but the connection is not lost and the quality will possibly improve either because the session will be served by another path, or because the link will recover. Of course, it is necessary to confirm whether the more reliable reconstruction of the degraded signal is of preferable quality than the less reliable reconstruction of the full-rate signal that would correspond to taking no action at all. In fact, the use of intelligent mapping of bits to symbols and/or the use of hierarchical compression [4] can be coupled with our technique to improve overall performance. The trade-offs involved in maintaining the best signal quality while adjusting adaptively the demodulation as well as choosing signal constellations are the subject of a separate study.

In the future we may consider multiple levels of transmission rate by alternatively switching among different Multiple-PSK schemes.

2.3 Detailed algorithm description

As mentioned above, each mobile node is aware of its connectivities to adjacent nodes as these are specified by the rules of the CE protocol. Furthermore, every mobile node maintains a *connectivity table* for the broadcast channel. This table is updated when a connection-request is placed and keeps an entry of the form

Connection-ID	Incoming link	Outgoing link	Transceiver-ID	Decoder status

with the connection ID number, the incoming and outgoing link and the transceiver ID that serves the session for connectivity purposes, plus a boolean variable that denotes what type of demodulator is being used. There is also a "flag" for each entry that specifies whether a CR has been acknowledged (and thus the session is in progress) or not.

We now describe in detail the procedures followed by the two logical phases of the algorithm.

2.3.1 Construction Phase

During the construction phase the algorithm utilizes three types of control packets: **Connection-Request (CR)**, **Acknowledgment (ACK)** and **Negative Acknowledgment (NACK)**. A CR consists of a source node identifier, a destination node identifier, a transmitting node identifier and a connection identifier which will be useful to distinguish the request from other requests placed by the same node, but also in the connectivity tables at the nodes. An ACK consists of the source identifier, the destination identifier, the connection identifier and the number of available transceivers in the full path to the DEST. A NACK consists of the source node identifier, the destination node identifier and the connection identifier.

- CR propagation

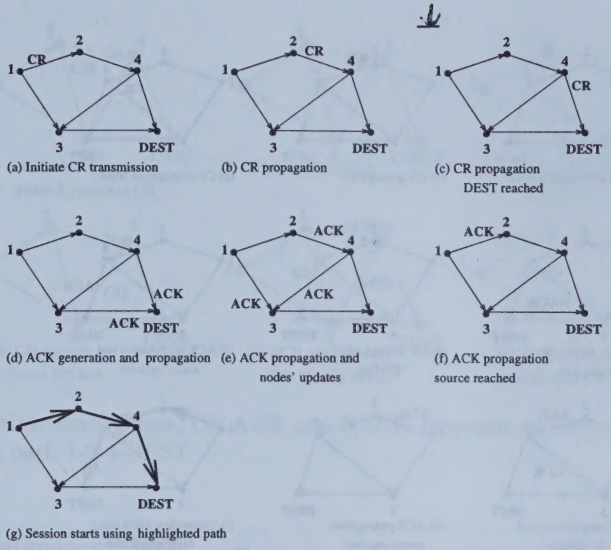


Figure 2: Construction phase, CR and ACK propagation

We assume that at some point in time a node, not adjacent to the DEST, will desire a connection with the DEST. Without loss of generality we assume that this node has already multiple DN links (see node 1 in example beginning with figure 2(a)) which implies according to the properties of the CE protocol that there exists at least one route to the DEST. Then the node selects one of the DN links over which it transmits a CR to its DN node. This selection is based on the information about transceivers' availability that has already been collected by previous control messages exchange and the node chooses to transmit among all DN nodes to the one that is in the path with the maximum number of available transceivers. If we are in an initial phase and no such information has already become available, the selection is made randomly.

Each node which receives a CR over an UP link checks if the connection can be routed through according to availability of resources. Mainly the node checks if a transceiver is available and in that case it reserves this transceiver and retransmits the connection-request to a DN node selected in the same fashion according to bandwidth information or randomly if no such information exists. We believe that there will be at least one DN node since this is assured by the rules of the CE protocol. In the case that no transceivers are available to accommodate the CR, the node broadcasts back to the upstream neighbor a NACK. So far we assumed that the node which received the connection request was not the destination node. If this is the case, it broadcasts backwards to all the upstream nodes an ACK.

- ACK propagation

  An ACK is generated during the construction phase by the destination node (see example, figure 2(c)(d)) upon reception of a CR packet that can be accommodated. Upon reception of an ACK, a node checks if that request was placed through it

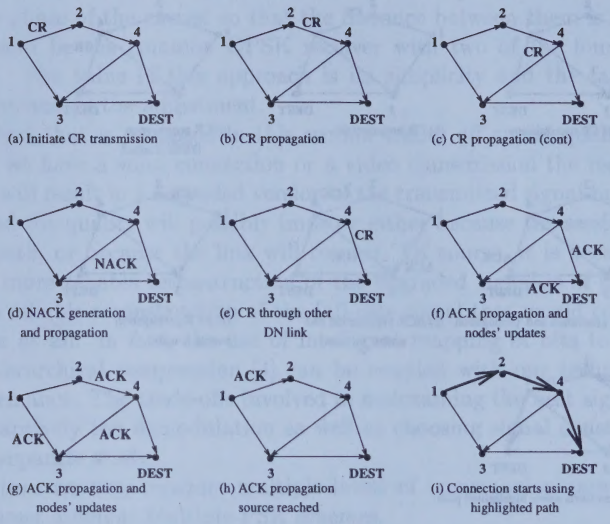


Figure 3: Construction phase, CR, NACK and ACK propagation

earlier by checking the connectivity table, and rebroadcasts the control packet to its upstream nodes by confirming acknowledgment of the specific request in the connectivity table. If the ACK is not for a connection placed through this node (node 3,figure 2(d), then it will be treated as an update control message. It contains the maximum number of transceivers available for use at a path from this node to the DEST. The node will rebroadcast the ACK to its upstream neighbors by replacing the transceivers number with the minimum between the number it has just received through the ACK and the number of its available transceivers. It is important to distinguish the double function of the ACK propagation which is simultaneously used as an acknowledgment packet and as an update message for availability of transceivers.

- NACK propagation

  A NACK is initiated by a node which has previously received a CR but has not the required number of transceivers available to accommodate the request (node 3, figure 3(d)). The NACK is transmitted backwards to the upstream node that transmitted the specific CR. The latter upon receiving the NACK checks whether it has other DN links over which it hasn't previously transmitted the same CR. This is achieved by marking "blocked" a link over which a NACK is received, so that the upstream node knows that its DN cannot accommodate the specific request. If no "unblocked" DN links exist, the NACK is broadcast backwards to the upstream node and the previously reserved transceivers for the specific connection become available again. If the NACK is received by the node which initiated the request, then this node has to repeat this phase through another DN link. If all DN links return a NACK, then the connection request is dropped.

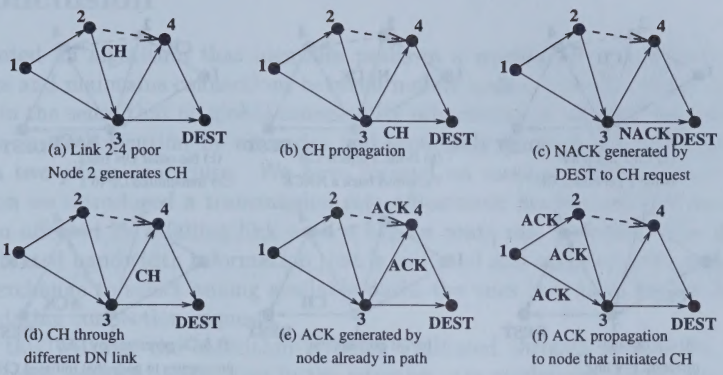


Figure 4: Maintenance phase, CH,ACK and NACK propagation because of soft failure of link 2-4 in path 1-2-4-DEST

2.3.2 Maintenance Phase

We describe now how the algorithm reacts to discover a new route and handoff a connection in progress that is temporarily served by the old path at a reduced rate. During the maintenance phase the algorithm utilizes one additional type of control packet, the **Connection-Handoff (CH)** that consists of the source node identifier, the destination node identifier, the transmitting node identifier and the connection identifier.

- CH propagation

  A node transmits a CH when one of its DN links that is being used by a session in progress demonstrates a poor quality. The CH is transmitted over one of the node's DN links, other than the one used by the session. In the construction phase description, the node would choose over which DN link to transmit a CR based on transceivers' availability in the whole path to the DEST. In the maintenance phase this is not certain that it will be a correct approach. Many CH's may fail because of lack of transceivers even though some of those transceivers will still be reserved for the same session and could be reused. Note that the connection is still in progress utilizing network resources. Hence, we decided to simulate network operation in two different cases. First by selecting one DN link according to bandwidth knowledge (with a high chance of failure) and second by selecting the DN link randomly.

  The CH propagation continues until an ACK is generated in reply to the CH (see example figure 4). An ACK can be generated either when the CH reaches the DEST or when it reaches a node that is already in the path that the session is utilizing (for instance node 4, figure 4(e)).

  A node upon receiving a CH checks if this connection ID is in its connectivity table. If so, the node will generate and transmit an ACK only if it is located in the segment of the path beyond the poor quality link. The node maintains a boolean variable that shows the status of the receiver. In our case, if the node that just received the CH is decoding in QPSK, it is in that segment of the path that has not been affected by the link degradation and the CH request will fail. If however,

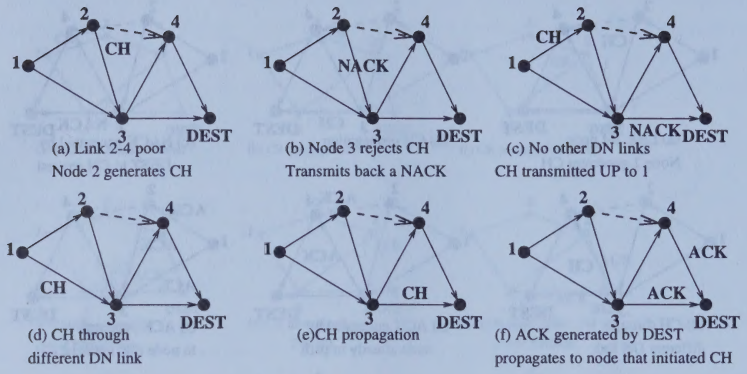


Figure 5: Maintenance phase, DN links exhausted

it is decoding in BPSK and it is not the node that initially generated the CH, it is located beyond the affected link and can generate an ACK since the rest of the path to the DEST can still be used. For example if the CH in figure 4 somehow reaches node 1 (by following a hypothetical path not shown in the figure), 1 should not accept the request even if it already serves the call. It should instead transmit the CH to node 3.

If a node has exhausted all its DN links without receiving any ACK, then it has to transmit the CH over the UP link of the session, and the upstream node that will receive it will repeat the CH procedure.

3 Simulation

The proposed algorithm is presently undergoing extensive simulation-based evaluation. For the simulation setup, we assume an underlying non-blocking channel access scheme which applies to the control channel, and allows as many nodes as possible to transmit provided no primary or secondary collisions occur.

We simulate the algorithm in a variety of network architectures in which node connectivities are not static but vary with time. We assume a random link state process taking place independently at each link, and we vary the rate of topological changes to examine algorithm performance in different environments.

We evaluate the algorithm by studying its reaction time to connectivity changes as a function of the rate of topological change, and by collecting statistics for percentage of connection-requests blocked or of sessions interrupted.

Simulations are in progress and any results will be published as soon as they become available.

4 Conclusion

We presented an algorithm that identifies paths in a wireless all-mobile network, and establishes and maintains connections between mobile nodes. The algorithm is fully distributed, in the sense that no global connectivity information is required from the nodes, and relies on the execution by all nodes and separately for each potential destination node of a two-phase procedure. We have focused on session-oriented service and for that reason we introduced a transmission rate adjustment mechanism that maintains a connection affected by a failing link until a bypass route can be found. The algorithm uses distributed bandwidth information that is collected and updated during the control message exchange to select among available paths the ones that have higher chances of accommodating connection requests.

As of this writing, the algorithm is being evaluated through simulation. We are also examining possible alternatives in the adaptive rate mechanism which may enhance performance. Finally, we are in the process of providing a theoretical validation to the algorithm, and of obtaining detailed proofs of its basic properties.

"The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory or the U.S. Government."

References

[1] M. S. Corson, and A. Ephremides, "A Distributed Routing Algorithm for Mobile Wireless Networks," *Wireless Networks*, Vol. 1, pp. 61-81, 1995.

[2] T. S. Rappaport, *Wireless Communications*, Prentice-Hall, 1996.

[3] J. G. Proakis, and M. Salehi, *Communications Systems Engineering*, Prentice-Hall, 1994.

[4] M. B. Pursley, "Signaling Methods and Routing Techniques for Mobile, Multimedia, Wireless Networks: An Overview of Recent Research Results," Workshop on New Techniques for Radio Communications and Wireless Networks, San Diego, CA, Sep. 1996.

[5] E. Perahia, C. Wang, C. J. Hansen, and G. J. Pottie, " Interference Suppression Using Antenna Arrays and Dynamic Power and Bit Allocation," *Proceedings of Interference Rejection and Signal Separation in Wireless Communications (IRSSWC) Symposium*, New Jersey Institute of Technology, Mar. 1996.

PREDICTIVE QUEUEING MULTIPLE ACCESS - A WIRELESS ATM PROTOCOL FOR MULTIMEDIA COMMUNICATION

Joseph Y. Hui
Dept of Electrical and Computer Engineering
Rutgers Univ., Piscataway NJ 08855-0909, email:jhui@ece.rutgers.edu

ABSTRACT

PQMA (Predictive Queueing Multiple Access) is a media access protocol for transporting speech, data, and video via indoor wireless channels. The 5-byte ATM header is modified to accommodate a multi-rate packet access protocol with ARQ capability. Similar to the ATM, flexible bandwidth assignment is achieved by the use of VCI. The ATM header contains a prediction field for the base station to schedule subsequent transmission from the mobile. Packet queues are maintained at both the base station and the mobile station. The protocol also has a built in capability for retransmission required for fading channels. Slow Frequency Hopping is also employed for combating frequency selective fading. The queueing and error performance of PQMA is analyzed and simulated, which demonstrates that PQMA can support multimedia communication for PCS.

I. Introduction

New applications are developing for wireless communications based on portable computers. Such terminals can provide functions such as mobile telephony, electronic mail, and file transfer. Higher bandwidth applications such as Web browsing, multimedia presentations, video on demand, and video conferencing are becoming popular on notebook computers. There is a need to use the wireless channel for broadband access to these computers. Such effort is broadly termed wireless multimedia or wireless ATM. Early proposals of wireless ATM is found is [1-3].

For wireline networks ATM was devised to allow flexible and on-demand allocation of transmission capacity for broadband networks. Also, ATM allows fast switching using Virtual Circuit Identifiers (VCI) and parallel switching fabric. More importantly, Quality of Service (QoS) can be carefully provisioned for different media types with vastly different burstiness, bit-rate, delay and error requirements. The ATM cell header is shown in fig 1.

For wireless networks supporting high bandwidth multimedia applications, it is desirable to maintain compatibility with the wireline ATM network, using similar notions of fixed cell size, VCI routing, and QoS provisioning. The advantages of ATM apply similarly for the wireless environment. However, the ATM header and protocol functions must be modified to reflect

major differences between the wireless and wireline networks.

The first difference is that the mobiles require the use of scheduling or contention algorithms in order to access the wireless channel. Such algorithms allow bursty and multi-rate establishment of multimedia virtual circuits.

The second difference is that a severely impaired wireless transmission environment requires the extensive use of both forward error correction coding and ARQ. Sporadic errors as well as changing channel condition may cause fluctuations in the required channel bit rate due to code adaptation and retransmission.

The third difference is that bandwidth is severely limited on the wireless channel. Multiple frequency bands may be necessary for high bandwidth applications.

We propose modifying the forward and reverse channel ATM header format to accommodate the different functionalities for a wireless channel. These functionalities include scheduling channel access and error control, both FEC and ARQ. Slow frequency hopping is adopted for providing more bandwidth and diversity protection against fading.

II. Protocol for Mobile to Base

We assume that ATM cells queue at both the base and the mobile for transmission. VPI/VCI translation occurs prior to the cell being put into the queue at the base station. The number of enqueued cells at either the base station or the mobile depends on the data generation rate of the connection, retransmission requests resulting from channel errors, and the rate at which cells are scheduled for transmission. Queue length could be one factor for scheduling transmission for connections in a microcell.

Each connection on the wireless ATM channel is labeled by 1 byte (fig 2), instead of the usual 3 1/2 byte VPI/VCI field for wireline ATM. Therefore as many as 256 virtual channels can be supported in a microcell. This should be sufficient since a microcell may have limited bandwidth.

The key concept for PQMA is the use of a prediction field in the cell header for explicit scheduling of the next cell for mobile to base communication. For each virtual circuit, a currently transmitted cell schedules the next cell by the parameter Prediction Field (PF), which represents the offset in term of slots from the current cell (fig 3). The offset depends on the current bit-rate of the circuit, and is approximately given by the total channel bit-rate divided by

the bit-rate of the virtual circuit after channel coding.

This offset may fluctuate from cell to cell due to bit-rate changes for a connection, retransmission needs, and fill status of the source buffer. The offset could be reduced if the source buffer is close to full, or increased if the source buffer is close to empty. Retransmissions tend to increase buffer occupancy and therefore reduces the offset.

This offset is coded as two subfields in the Prediction Field as frame offset and slot offset. The wireless channel is divided into frames. Each frame contains 32 slots. Each slot may carry a 53 byte cell. The 7-bit frame offset refers to the frame location of the next scheduled cell. The 5-bit slot offset refers to the slot location in the referenced frame.

One byte of the 5-byte ATM header is used for error control functions. First, a 4-bit sequence number (SEQ) represents cyclically the cell sequence number. This is used for selective repeat of errored transmission. A 3-bit code rate field represents the error control code being used. A 1-bit New field is set to 1 if the cell is the first cell of a connection. We shall describe in detail the error control mechanism in a later section.

The remaining 1 1/2 byte is identical to that for wireline ATM, namely a 3-bit Payload Type Identifier (PTI), a 1-bit Cell Loss Priority (CLP), and an 8-bit Header Error Check (HEC) field.

III. Scheduling Protocol at the Base Station

Scheduling is performed primarily at the base station. A schedule table (fig 4) is kept, which records the VCI scheduled for slots sequentially. Besides the VCI, the slot entry in the table also records the PTI of the VCI and a delay value to be explained later. A pointer PTR is used to reference the current frame in the table.

When an ATM cell arrives, the base examines the entry in the table indexed by PTR plus the offset indicated by the PF field. If the slot is not scheduled, then the VCI, the PTI, and delay=0 is entered in the empty entry. If the entry is occupied, then a contention arises. The contention is resolved first by using the PTI field. If the PTI happens to be identical, then the contending party with a larger delay wins. The losing party then contends for the next slot with delay increased by one.

At the beginning of each frame, the base broadcast the frame schedule for the next (PTR+1) frame. This broadcast takes up one slot in frame PTR, listing sequentially the VCI entries for frame PTR+1 in the schedule table. For a frame size of 32 slots, 32 bytes are used for indicating

the VCIs.

Since the scheduling is explicit, the base expects the mobile to send a cell in the scheduled slot. If for some reasons the reception fails, the base would know immediately and request a retransmission. The requests could be made in the next frame using the broadcast slot. The negative acknowledgements (NAKs) are represented by a 1 byte VCI and a 4-bit SEQ. The mobile then selectively repeats the cell being negatively acknowledged. Excessive number of NAKs for a VCI may triggers an exception control condition, which indicates either a channel or protocol failure.

Two additional variations of this basic scheduling algorithm could be made. First, the unscheduled slots could be used by mobiles using random access methods such as Aloha. The unscheduled slots could be used for selective repeat or for easing temporary backlog in the buffer of a mobile. If the reception at the base is successful, positive acknowledgement (ACKS) could be sent in the broadcast slot in the next frame, indicating the VCI and SEQ of the successful contender.

Second, scheduling can also be performed in the frequency domain besides the time domain. This is particularly useful if slow frequency hopping is employed. The broadcast slot now contains the list of VCIs for all slots referenced by both time and frequency within a frame duration.

Besides the broadcast slot, protocol information is also carried by the ATM headers of the forward (base to mobile) channels. For base to mobile communications, the base station can easily schedule slots for virtual circuits. The ATM header contains a 1-byte VCI field. The mobile examines the VCI field to see if the ATM cell is marked for the mobile. Instead of inband transport of the VCI within the ATM cell, out-of-band transport of the VCI in the broadcast slot can also be used. The 1 1/2 byte PF field is not necessary for the forward channel. The rest of the ATM header remains the same as the reverse channel.

IV. ARQ and FEC Methodologies

In this section, we examine how errors may arise and be dealt with. Errors may affect the functioning of the protocol described, or corrupt the data transport directly.

Protocol information errors can arise if the broadcast slot, the VCI, PF, or NAK are corrupted. Since scheduling is explicit, the base station could detect such errors if the mobile fails to send the scheduled cell. NAKs are sent immediately under such conditions. For lost PFs, the base can

use previous values of PF for the virtual circuit, since the value of PF need not be exact. We believe that our scheme is robust to occasional protocol information errors of various types. If such errors are persistent, the base triggers an exception condition.

Several methods are available for handling corrupted data transport. As mentioned earlier, the 48-byte information payload may contain a CRC itself for detecting errors within the cell. A NAK could be sent requesting a retransmission if an error is detected.

A variation of this NAK scheme could be implemented for powerful error correction. We call the scheme Progressive ARQ Reed-Solomon Coding (PQRS coding). The code letters are elements of the Galois Field GF(64). Each letter is therefore 6 bit long. Consider using a (256,56) RS code. This code is broken into 4 sub-blocks, with each sub-block being a (64,56) RS code. Each sub-block codeword is therefore of length 64 x 6 bits or 48 bytes, fitting nicely within the ATM cell payload. A block of 56 x 6 bits or 42 bytes is encoded by the (64,56) RS code and sent in an ATM cell. If the base fails to decode correctly, a NAK is sent to the mobile. The mobile then sends the next (64,56) sub-block. The base, using both cells, tries to decode the resulting (128,56) RS code. If this also fails, the base asks for the next sub-block, etc. We have simulated this progressive scheme and found that throughput and error performance to be excellent for the fading channel.

The 3-bit code rate field represents the coding method and rate of the FEC used. If convolutional code is used, a variety of code rates such as 2/3, 1/2, 1/3, and 1/4 could be used. If errors occur frequently, the code rate may adjust to the next lower rate.

We propose this wireless ATM scheme to be used in conjunction with Slow Frequency Hopping (SFH) with moderate interleaving for the purpose of combating frequency selective fading. We have also explored the possibilities of adaptive transmission baud-rate, which may allow higher transmission rate if a direct line of sight propagation condition exists.

V. Experimental Results

Extensive simulation is performed for the following system. For this short paper, we use the following example. We assume for indoor radio that a total of 20 Mb/s capacity is available. Slow Frequency hopping with rate 1/2 convolutional coding and interleaving are employed. We assume three kinds of services, namely 16 Kb/s speech, 1.5 Mb/s video, and Web data at 1 Mb/s. These bit rates are expanded by 2 upon convolutionally coded. An active terminal use any one of the three services at one time with probability 0.4, 0.2, 0.4 respectively. Call admission control can be determined by examining the region where delay for these services is acceptable. For this

example, we examine throughput and delay versus the number of active terminals with the above service type probabiities.

Figure 5 illustrates the throughput as more active terminals are admitted. Throughput saturates around 12 terminals. The queueing delay suffered by video according to the PQMA scheme with earliest deadline first scheme is shown in figure 6.

The above simulation is performed under the assumption that the channel is error free. To provide input to the simulation when the channel is not error free, or to see under what regime we may assume an error free condition, we have simulated also the effect of coding, interleaving and channel equalization on a Raleigh faded channel with severe intersymbol interference. Figure 7 shows the BER for a root mean square delay spread of 0.4, a convolutional channel coding rate 1/2 with constraint length 5, and hopping among 1, 3, 5, or 7 channels. A maximum likelihood decoder for both ISI and convolutional coding via Viterbi decoding is implemented. It shows substantial diversity effect when 5 or more hopping channels are used. In other plots, we have also demonstrated that having a large delay spread (say 0.4 instead of 0.1) in fact provides an additional diversity effect, with reduced BER.

VI.Conclusion.

We propose a wireless ATM protocol which provides true on demand bandwidth allocation using the notion of timed reservation via PQMA. Preliminary results show that throughput and delay are acceptable for indoor applications. Further work will be performed on how delay performance could be fine tuned through more advance scheduling techniques.

References

1. D. Raychaudhuri, N.D. Wilson, "ATM-Based Transport Architecture for Multiservices Wireless Personal Communication Networks," IEEE JSAC, vol. 12 Oct. 1992.
2. A.S. Acampora, M. Nagshineh, "An Architecture and Methodology for Mobile-Executed Handoff in Cellular ATM Networks,"IEEE JSAC, vol 12 Oct 1994
3. E. Ayanoglu, K. Eng, M. Karol, "Wireless ATM: Limits, Challenges, and Proposals," IEEE Personal Communications, Vol. 3, no 4. August 1996.

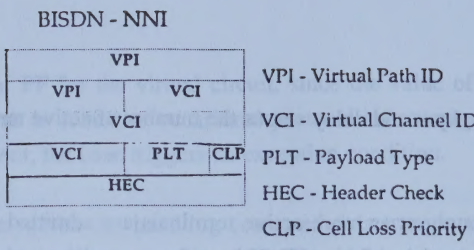


Fig. 1 Wireline ATM Cell Header

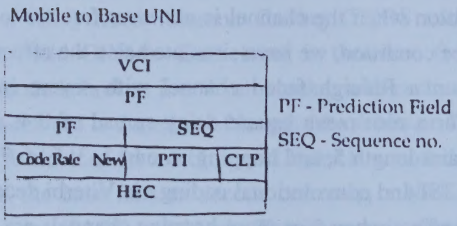


Fig. 2 Wireless ATM Cell Header

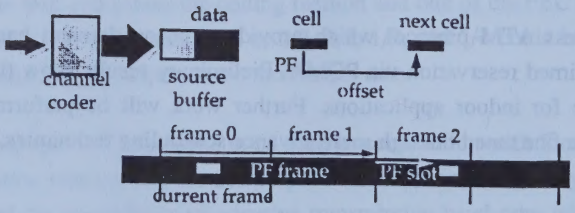


Fig. 3 Offset and its Representation

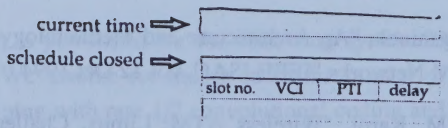


Fig. 4 Schedule Table at Base

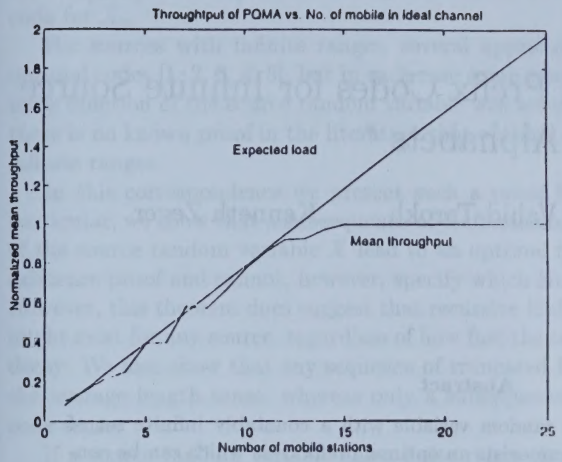

Throughput of PQMA vs. No. of mobile in ideal channel
Normalized mean throughput
Number of mobile stations
Expected load
Mean throughput

Fig. 5 Throughput for PQMA

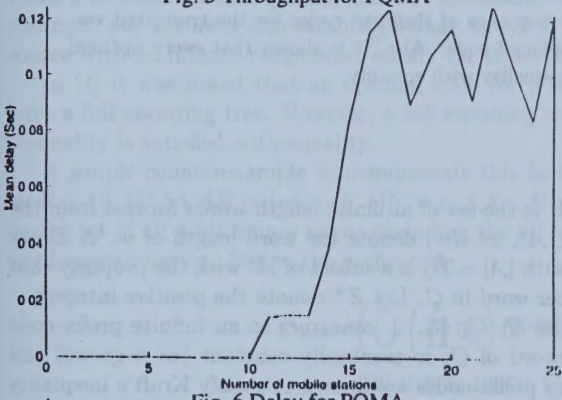


Fig. 6 Delay for PQMA

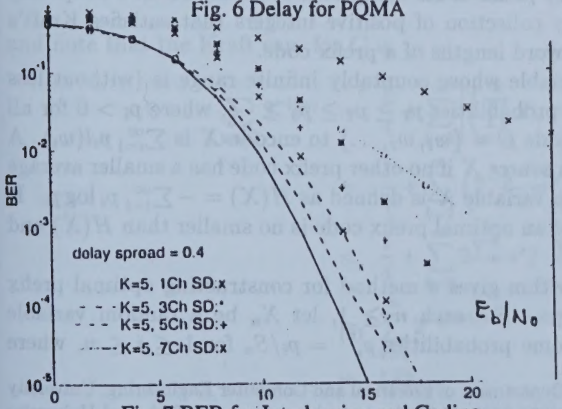


Fig. 7 BER for Interleaving and Coding

Existence of Optimal Prefix Codes for Infinite Source Alphabets *

Tamás Linder Vahid Tarokh Kenneth Zeger

Abstract

It is proven that for every random variable with a countably infinite set of outcomes and finite entropy there exists an optimal prefix code which can be constructed from Huffman codes for truncated versions of the random variable, and that the average lengths of any sequence of Huffman codes for the truncated versions converge to that of the optimal code. Also, it is shown that every optimal infinite code achieves Kraft's inequality with equality.

1 Introduction

An *alphabet* $\mathcal{A}$ is a finite set and $\mathcal{A}^*$ is the set of all finite length words formed from the elements of $\mathcal{A}$. For each word $w \in \mathcal{A}^*$, let $l(w)$ denote the word length of w. A D-ary *prefix code* C over an alphabet $\mathcal{A}$ (with $|\mathcal{A}| = D$) is a subset of $\mathcal{A}^*$ with the property that no word in C is the prefix of another word in C. Let $\mathcal{Z}^+$ denote the positive integers.

A sequence of D-ary prefix codes $C_1, C_2, C_3, \ldots$ *converges* to an infinite prefix code C if for every $i \geq 1$, the i^{th} codeword of C_n is eventually constant (as n grows) and equals the i^{th} codeword of C. D-ary prefix codes are known to satisfy Kraft's inequality $\sum_{w \in C} D^{-l(w)} \leq 1$. Conversely any collection of positive integers that satisfies Kraft's inequality corresponds to the codeword lengths of a prefix code.

Let X be a source random variable whose countably infinite range is (without loss of generality) $\mathcal{Z}^+$, with respective probabilities $p_1 \geq p_2 \geq p_3 \geq \cdots$, where $p_i > 0$ for all i. The average length of a prefix code $C = \{w_1, w_2, \ldots\}$ to encode X is $\sum_{i=1}^{\infty} p_i l(w_i)$. A prefix code C is called *optimal* for a source X if no other prefix code has a smaller average length. The entropy of the random variable X is defined as $H(X) = -\sum_{i=1}^{\infty} p_i \log p_i$. It is known that the average length of an optimal prefix code is no smaller than $H(X)$ and is smaller than $H(X) + 1$.

The well-known Huffman algorithm gives a method for constructing optimal prefix codes for sources with finite ranges. For each $n \geq 1$, let X_n be a random variable with a finite range and with outcome probabilities $p_i^{(n)} = p_i / S_n$ for $1 \leq i \leq n$, where

*T. Linder and K. Zeger are with the Department of Electrical and Computer Engineering, University of California, San Diego, La Jolla, CA 92093-0407. T Linder is on leave from the Technical University of Budapest, Hungary. (email: {linder, zeger}@code.ucsd.edu). V. Tarokh is with the AT&T Laboratories, 600 Mountain Avenue Murray Hill, N.J. 07974-0636. (email: tarokh@research.att.com).
This research was supported in part by the National Science Foundation.

$S_n = \sum_{j=1}^{n} p_j$. A *D-ary truncated Huffman code* of size n for X is defined to be a Huffman code for X_n.

For sources with infinite ranges, several approaches have been taken to construct optimal codes [1, 2, 3, 4, 5], but in each case some condition on the tail of the probability mass function of the source random variable was assumed. To the best of our knowledge there is no known proof in the literature that optimal codes always exist for sources with infinite ranges.

In this correspondence we present such a proof for sources with finite entropy. In particular, we show that a subsequence of Huffman codes designed for truncated versions of the source random variable X lead to an optimal infinite code for X. We provide an existence proof and cannot, however, specify which Huffman code subsequence is needed. However, this theorem does suggest that recursive Huffman code construction algorithms might exist for any source, regardless of how fast the tails of its probability mass function decay. We also show that any sequence of truncated Huffman codes indeed converges in the average length sense, whereas only a subsequence is guaranteed to converge in the code sense.

If a source random variable has a finite range then an optimal binary code satisfy's Kraft's condition with equality, but not necessarily for D-ary codes when $D \geq 3$. In contrast, our theorem also establishes that for all $D \geq 2$ an optimal D-ary code for a source with an infinite range must satisfy the Kraft inequality with equality.

In [4] it was noted that an optimal code for a source with an infinite range must have a full encoding tree. However, a full encoding tree does not guarantee that Kraft's inequality is satisfied with equality.

A simple counterexample to demonstrate this fact for $D = 2$ is given next. For any $A, B \subset \{0,1\}^*$ let $AB = \{ab \in \{0,1\}^* : a \in A, b \in B\}$. For $n \geq 0$, let $T_n = \{0,1\}^n \setminus \{0^n\}$ be the set of all n-bit binary words excluding the all zeros word, and let $\prod$ denote binary word concatenation. Define the prefix code

$$\begin{aligned} C &= \left(\bigcup_{k=2}^{\infty} \left(\prod_{n=2}^{k} T_n \right) 0^{k+1} \right) \cup \{00\} \\ &= \{00, 01000, 10000, 11000, \ldots\} \end{aligned}$$

and note that the Kraft sum for C is

$$\begin{aligned} \sum_{w \in C} 2^{-l(w)} &= \frac{1}{4} + \sum_{k=2}^{\infty} \left| \left(\prod_{n=2}^{k} T_n \right) 0^{k+1} \right| 2^{-\sum_{i=2}^{k+1} i} \\ &= \frac{1}{4} + \sum_{k=2}^{\infty} \left(2^{-\sum_{i=2}^{k+1} i} \right) \prod_{n=2}^{k} (2^n - 1) \\ &< \frac{1}{4} + \sum_{k=2}^{\infty} 2^{\sum_{i=2}^{k} i} 2^{-\sum_{i=2}^{k+1} i} \\ &= \sum_{k=1}^{\infty} 2^{-(k+1)} \\ &= 1/2. \end{aligned}$$

Thus the Kraft inequality is strict in this case and it is easy to see that the encoding tree of the code C is full.

2 Main Result

Theorem 1 *Let X be a random variable with a countably infinite set of possible outcomes and with finite entropy. Then for every $D > 1$, the following hold:*

(I) *There exists a sequence of D-ary truncated Huffman codes for X which converges to an optimal code for X.*

(II) *The average codeword lengths in any sequence of D-ary truncated Huffman codes converge to the minimum possible average codeword length for X.*

(III) *Any optimal D-ary prefix code for X must satisfy the Kraft inequality with equality.*

Proof For each $n \geq 1$, let C_n be a D-ary truncated Huffman code of size n for X, and denote the sequence of n codeword lengths of C_n (followed by zeros) by $l^{(n)} = \{l_1^{(n)}, l_2^{(n)}, \ldots, l_n^{(n)}, 0, 0, 0, \cdots\}$. Let $\mathcal{F}$ denote the set of all sequences of positive integers. For each n, the average length $\sum_{i=1}^{\infty} l_i^{(n)} p_i^{(n)}$ of Huffman code C_n is not larger than $H(X_n)+1$, where the entropy of X_n is

$$H(X_n) = -\frac{1}{S_n}\sum_{i=1}^{n} p_i \log p_i - \log\frac{1}{S_n} \to H(X) \text{ as } n \to \infty,$$

since $S_n = \sum_{i=1}^{n} p_i \to 1$ as $n \to \infty$. Hence $H(X_n) + 1 \leq H(X) + 2$, for n sufficiently large, and it is easy to see that $l_i^{(n)} \leq (H(X)+2)/p_i$ for n sufficiently large.

Thus, for each i, the sequence of codeword lengths $\{l_i^{(1)}, l_i^{(2)}, l_i^{(3)}, \cdots\}$ is bounded and therefore the corresponding sequence of codewords can only take on a finite set of possible values. Hence, for each i, there is a convergent subsequence of codewords. We conclude (using a minor modification of [6, Theorem 7.23]) that there exists a subsequence of codes $C_{n_1}, C_{n_2}, C_{n_3}, \ldots$ that converges to an infinite code $\hat{C}$. Clearly $\hat{C}$ is a prefix code since it is a limit of finite Huffman codes. Furthermore the subsequence $\{l^{(n_k)}\}$, of elements of $\mathcal{F}$, converges to a sequence $\hat{l} = \{\hat{l}_1, \hat{l}_2, \cdots\} \in \mathcal{F}$, in the sense that for each $i \in \mathcal{Z}^+$, the sequence $l_i^{(n_k)}$ converges to $\hat{l}_i$.

To show the optimality of $\hat{C}$, let $\lambda_1, \lambda_2, \lambda_3, \ldots$ be the codeword lengths of an arbitrary prefix code. For every k, there exists a j such that $\hat{l}_i = l_i^{(n_m)}$ for every $i \leq k$ provided that $m \geq j$. Thus for all $m \geq j$ and $k \leq n_m$, the optimality of Huffman codes implies

$$\sum_{i=1}^{k} p_i \hat{l}_i = \sum_{i=1}^{k} p_i l_i^{(n_m)} \leq \sum_{i=1}^{n_m} p_i l_i^{(n_m)} = S_{n_m} \sum_{i=1}^{n_m} p_i^{(n_m)} l_i^{(n_m)} \leq S_{n_m} \sum_{i=1}^{n_m} p_i^{(n_m)} \lambda_i. \tag{1}$$

Therefore, by letting first $m \to \infty$ and then $k \to \infty$, we obtain

$$\sum_{i=1}^{\infty} p_i \hat{l}_i \leq \sum_{i=1}^{\infty} p_i \lambda_i.$$

This implies that the infinite code $\hat{C}$ is optimal.

To prove part (II) of the theorem, notice that by the optimality of Huffman codes

$$\sum_{i=1}^{n} p_i l_i^{(n)} = S_n \sum_{i=1}^{n} p_i^{(n)} l_i^{(n)} \leq S_n \sum_{i=1}^{n} p_i^{(n)} l_i^{(n+1)} = \sum_{i=1}^{n} p_i l_i^{(n+1)} \leq \sum_{i=1}^{n+1} p_i l_i^{(n+1)}.$$

The sequence $\sum_{i=1}^{n} p_i l_i^{(n)}$ is thus an increasing sequence which is bounded above by $H(X)+2$ and has a limit L. It follows from (1) that

$$\sum_{i=1}^{\infty} p_i \hat{l}_i \leq \lim_{m\to\infty} \sum_{i=1}^{n_m} p_i l_i^{(n_m)} = \lim_{n\to\infty} \sum_{i=1}^{n} p_i l_i^{(n)}$$

Next by the optimality of Huffman codes

$$\lim_{n\to\infty} \sum_{i=1}^{n} p_i l_i^{(n)} = \lim_{n\to\infty} S_n \sum_{i=1}^{n} p_i^{(n)} l_i^{(n)} \leq \lim_{n\to\infty} S_n \sum_{i=1}^{n} p_i^{(n)} \hat{l}_i = \lim_{n\to\infty} \sum_{i=1}^{n} p_i \hat{l}_i = \sum_{i=1}^{\infty} p_i \hat{l}_i.$$

Thus

$$\lim_{n\to\infty} \sum_{i=1}^{n} p_i^{(n)} l_i^{(n)} = \lim_{n\to\infty} \frac{1}{S_n} \sum_{i=1}^{n} p_i l_i^{(n)} = \sum_{i=1}^{\infty} p_i \hat{l}_i.$$

This proves the second part of the theorem.

Next we prove part (III) of the theorem. Let the codeword lengths of an optimal code be denoted $l_1 \leq l_2 \leq l_3 \leq \cdots$, and assume to the contrary that the Kraft inequality is strict, i.e. $\sum_i D^{-l_i} < 1$. Let $\delta = 1 - \sum_i D^{-l_i} > 0$. Then there exists a positive integer k such that $D^{-l_i} < \delta$ for all $i \geq k$. Let j be an integer such that $l_j > l_k$. Define a collection of integers $\hat{l}_1, \hat{l}_2, \cdots$ such that $\hat{l}_i = l_i$ for all $i \neq j$ and such that $\hat{l}_j = l_k$. Then

$$\sum_{i=1}^{\infty} D^{-\hat{l}_i} = \sum_{i=1}^{\infty} D^{-l_i} - D^{-l_j} + D^{-l_k} < \sum_{i=1}^{\infty} D^{-l_i} + \delta = 1.$$

Thus the integers $\hat{l}_1, \hat{l}_2, \cdots$ satisfy Kraft's inequality, so that there exists a prefix code having them as codeword lengths. Since $\hat{l}_j < l_j$, such a prefix code will have a strictly smaller average codeword length for X than the optimal code whose codeword lengths are $l_1, l_2, \cdots$. This is a contradiction. □.

References

[1] J. Abrahams, "Huffman-type codes for infinite source distributions," *Journal of the Franklin Institute,* vol. 331B, no. 3, pp. 265-271, 1994.

[2] R. A. Gallager and D. C. Van Voorhis, "Optimal Source Coding for Geometrically Distributed Integer Alphabets," *IEEE Trans. Inform. Theory*, vol. 21, no. 2, pp. 228-230, 1975.

[3] P. A. Humblet, "Optimal Source Coding for a Class of Integer Alphabets," *IEEE Trans. Inform. Theory*, vol. 24, no. 1, pp. 110-112, 1978.

[4] B. Montgomery and J. Abrahams, "On the Redundancy of Optimal Binary Prefix-Condition Codes for Finite and Infinite Sources," *IEEE Trans. Inform. Theory*, vol. 33, no. 1, pp. 156-160, 1987.

[5] A. Kato, T. S. Han, and H. Nagaoka, "Huffman Coding with Infinite Alphabet," *IEEE Trans. Inform. Theory,* vol. 42, no. 3, pp. 977-984, May 1996.

[6] W. Rudin, *Principles of Mathematical Analysis*, Third Edition, McGraw-Hill Co., New York, 1976.

Identification plus Transmission over Channels with Perfect Feedback *†

Sivarama Venkatesan
Dept. of Electrical Engg.
Cornell University
Ithaca, NY 14853

Venkat Anantharam
Dept. of EECS
Univ. of California
Berkeley, CA 94720

Abstract

Suppose that a transmitter is connected to N receivers by a discrete memoryless channel W, with Shannon capacity $C > 0$, whose output is seen by all the receivers *and* the transmitter, i.e., there is perfect feedback. In n channel uses, the transmitter wishes to send one of M messages to *one* of the N receivers. It is required that, with high probability, the intended recipient should decode the message correctly, and any other receiver should recognize that the message is not intended for it. We prove that M and N can then *simultaneously* grow as $\exp[nR_1 - o(n)]$ and $\exp\{\exp[nR_2 - o(n)]\}$, respectively, for all $R_1 \leq C$ and $R_2 \leq \max_{P:I(P;W)\geq R_1} H(PW)$, provided randomized encoding is allowed. If the encoding must be deterministic, we prove that all (R_1, R_2) satisfying $R_1 \leq C$ and $R_2 \leq R_1 + \max_{P:I(P;W)\geq R_1} H(W|P)$ are achievable. In both cases, we establish these rate regions as optimal by proving "strong" converses.

1 Introduction

Consider the following communication problem: a transmitter is connected to N different receivers by a single discrete memoryless channel (DMC) W, whose output can be observed by all the receivers. The transmitter wishes to send one of M messages to *one* of the N receivers, but the receivers do not know beforehand who the intended recipient is. The transmitter must therefore encode both the *address* and the *message* in a codeword, and send it across the channel (the address is just the index of the intended recipient). Each receiver must then observe the channel output and decide whether the message is intended for it, and, if so, decode it. The requirements are that (a) the intended recipient should decode the transmitted

*Research supported by NSF IRI 9005849, IRI 9310670, NCR 9422513, and the AT&T Foundation.

†Address all correspondence to the first author (currently at U.C. Berkeley as an Exchange Scholar): 341, Cory Hall, Dept. of EECS, Univ. of California, Berkeley, CA 94720.

message correctly with high probability, and (b) any other receiver should recognize with high probability that the message is not intended for it (these probabilities are averaged over the M possible messages). Receivers other than the intended one do not need to know either the message or the actual recipient. The question of interest, then, is how fast M and N can simultaneously grow with the number of channel uses permitted to the transmitter.

If there is only one receiver, i.e. $N = 1$, then this is just the classical *data transmission* problem of information theory. In n uses of the channel, the transmitter can then reliably send one of $M = \exp[nR - o(n)]$ messages to the receiver, provided $R \leq C$, where $C = \max_P I(P; W)$ is the Shannon capacity of the channel W — the maximum mutual information between the channel input and output, over all input probability distributions P. (Throughout the paper, all exponentials and logarithms will be to the base e.)

On the other hand, if $M = 1$, we have the *identification* problem introduced by Ahlswede and Dueck [2]. Here, we could think of the transmitter as simply sending an alarm or "identifying" signal to one of the N receivers (there is no message to be transmitted). In [2], it was shown that one of $N = \exp\{\exp[nR - o(n)]\}$ receivers (*doubly* exponential in n) could be reliably identified in n channel uses, *provided* randomization was allowed in the encoding process. The identification capacity of the channel, defined as the maximum achievable (second-order) rate R in this situation, was shown to equal its Shannon capacity C! In contrast with the transmission problem, randomization in the encoding is crucial here, in the sense that it is impossible to achieve any positive second-order rate deterministically, i.e., the deterministic identification capacity is zero.

The general problem of simultaneous *identification and transmission* was studied by Han and Verdu [4], as a variant of the identification problem with potential applications in multiuser communication. The straightforward "time-sharing" solution to this problem would be to encode the message in $(1 - \alpha)n$ symbols using a classical transmission code, and encode the address in a header of αn symbols using an identification code (n is the number of channel uses). This scheme would permit $M = \exp[n(1-\alpha)C - o(n)]$ and $N = \exp\{\exp[n\alpha C - o(n)]\}$, provided randomization was allowed in the address encoding. However, it is possible to do better. In [4], it was shown that in fact $M = \exp[nC - o(n)]$ and $N = \exp\{\exp[nC - o(n)]\}$ are *simultaneously* achievable, if the address and message are *jointly* encoded using an "identification plus transmission" code, instead of separately as above; moreover, the joint encoding does *not* require randomization.

In this paper, we study an analogous identification plus transmission problem when the transmitter has *perfect and instantaneous feedback* from the output of the DMC connecting it to the receivers. It is well known that feedback does not increase the transmission capacity of a DMC. In marked contrast, feedback can have a dramatic effect on its identification capacity. In [1], it was shown that feedback allows positive second-order identification rates even when the encoding has to be deterministic. In fact, the deterministic feedback identification capacity of a DMC W with positive Shannon capacity was shown to be $\max_x H[W(\cdot|x)]$ — the maximum, over all input symbols x, of the conditional output entropy when x is transmitted. Allowing randomized encoding was shown to increase the feedback identification

capacity to $\max_P H(PW)$ — the maximum unconditional output entropy, over all input probability distributions P.

Here, we determine the region of all rate-pairs (R_1, R_2) such that the transmitter can reliably send one of $M = \exp[nR_1 - o(n)]$ messages to one of $N = \exp\{\exp[nR_2 - o(n)]\}$ receivers across the DMC W, in the presence of feedback. As in [1], we consider both the case where randomized encoding is allowed, and the case where the encoding must be deterministic. The problem is formulated more precisely in Section 2, and the results are stated in Theorem 2.1. The identification theorems of [1] can be viewed as special cases of these results, obtained by setting the transmission rate requirement R_1 to zero. As in [1], the converses proved here are "strong." As a by-product of these converses, we also have a new proof of the strong converse to Shannon's coding theorem for DMCs with feedback, a result first proved by Kemperman [5].

2 Statement of problem and results

The DMC connecting the transmitter and the receivers is assumed to have finite input and output alphabets $\mathcal{X}$ and $\mathcal{Y}$, respectively, and transition probability function $W = \{W(y|x) : x \in \mathcal{X}, y \in \mathcal{Y}\}$. In the presence of feedback, the transmitter can choose its channel input at each step of communication based on the outputs from all previous steps. Accordingly, we define an *n-step feedback function* as a vector $f = (f_1, \ldots, f_n)$, where $f_k : \mathcal{Y}^{k-1} \to \mathcal{X}$. When the transmitter uses this feedback function for communication, it sends the symbol $X_k = f_k(Y^{k-1})$ in step k, $1 \le k \le n$, where Y^{k-1} is the sequence of channel outputs in the first $k-1$ steps. The corresponding probability that $Y^n = y^n$, denoted $Q_f(y^n)$, equals $W^n(y^n \,|\, f(y^n))$, where $f(y^n)$ is the $x^n \in \mathcal{X}^n$ given by $x_k = f_k(y^{k-1})$, $1 \le k \le n$. Also, the probability that $(X^n, Y^n) = (x^n, y^n)$, denoted $W_f(x^n, y^n)$, equals $Q_f(y^n)$ if $x^n = f(y^n)$ and 0 otherwise.

An *n-step feedback strategy* F for the channel W is defined as a probability distribution (p.d.) on $\mathcal{F}_n$, the set of all n-step feedback functions. To communicate according to the strategy F, the transmitter randomly chooses $f \in \mathcal{F}_n$ with distribution F, and then uses it as described above, to decide its channel input at each step. The probability that $(X^n, Y^n) = (x^n, y^n)$ under the strategy F, denoted $W_F(x^n, y^n)$, is then $\sum_{f \in \mathcal{F}_n} F(f) W_f(x^n, y^n)$, and the corresponding marginal probability that $Y^n = y^n$, denoted $Q_F(y^n)$, is $\sum_{f \in \mathcal{F}_n} F(f) Q_f(y^n)$.

A general strategy, as defined above, is allowed to use randomization. The strategy F is called *deterministic* if $F(f) = 1$ for some $f \in \mathcal{F}_n$. Clearly, such a strategy does not require any randomization.

Definition 2.1 *An (n, N, M, λ, μ) identification plus transmission (IT) code is a collection $\{(F_{a,m}, \mathcal{D}_{a,m}) : (a, m) \in [N] \times [M]\}$,[1] where $F_{a,m}$ is an n-step strategy, $\mathcal{D}_{a,m} \subseteq \mathcal{Y}^n$, and for each $a \in [N]$:*

(1) $\mathcal{D}_{a,m} \cap \mathcal{D}_{a,m'}$ is empty if $m \neq m'$;
(2) $(1/M) \sum_m Q_{F_{a,m}}(\mathcal{D}_{a,m}) > 1 - \lambda$;

[1] If J is a positive integer, then $[J]$ will denote the set $\{1, 2, \ldots, J\}$.

(3) $(1/M)\sum_m Q_{F_{a,m}}(\mathcal{D}_{a'}) < \mu$ *for all* $a' \neq a$, *where* $\mathcal{D}_{a'} = \bigcup_m \mathcal{D}_{a',m}$.

The code is called deterministic if all the strategies $F_{a,m}$ *are deterministic.*

The interpretation of the above code is as follows: if the transmitter wishes to send message m to receiver a, it communicates according to the strategy $F_{a,m}$. After n steps, receiver a decides that it is indeed the intended recipient if the received sequence Y^n falls in $\mathcal{D}_a = \bigcup_m \mathcal{D}_{a,m}$; in this case, because of Condition (1) in the definition, there is a unique m such that $Y^n \in \mathcal{D}_{a,m}$, and it takes this m as the transmitted message. Otherwise, i.e. if $Y^n \notin \mathcal{D}_a$, receiver a decides that the message is not intended for it. Condition (2) guarantees that the intended recipient decodes the transmitted message correctly with probability greater than $1-\lambda$, while Condition (3) guarantees that any other receiver wrongly decides it is the recipient with probability less than μ. Note that these probabilities are averaged over the M possible messages.

The rate-pair (R_1, R_2) will be called (λ, μ)-*achievable* if there exists a sequence of $(n, N_n, M_n, \lambda, \mu)$ IT codes such that

$$\liminf n^{-1} \log M_n = R_1 \quad \text{and} \quad \liminf n^{-1} \log\log N_n = R_2.$$

If this sequence can be chosen to be deterministic, then (R_1, R_2) will be called *deterministically* (λ, μ)-*achievable.*

Theorem 2.1 (Main Theorem) *Assume that the discrete memoryless channel* W *has positive Shannon capacity* $C = \max_P I(P;W)$. *Then, for any* (λ, μ) *satisfying* $\lambda > 0$, $\mu > 0$, *and* $\lambda + \mu < 1$, *the rate-pair* (R_1, R_2) *is* (λ, μ)-*achievable if and only if*

$$R_1 \leq C \quad \text{and} \quad R_2 \leq \max_{P: I(P;W) \geq R_1} H(PW),$$

and is deterministically (λ, μ)-*achievable if and only if*

$$R_1 \leq C \quad \text{and} \quad R_2 \leq R_1 + \max_{P: I(P;W) \geq R_1} H(W|P).$$

The assumptions $\lambda > 0$ and $\mu > 0$ are of course reasonable. We also need the assumption $\lambda + \mu < 1$ in order to get meaningful results; if $\lambda + \mu > 1$, it can be shown that arbitrarily high identification rates R_2 are achievable. The assumption that $C > 0$ precludes trivialities of the opposite kind; if $C = 0$, it can be shown that there does not exist any (n, N, M, λ, μ) IT code with $N > 1$ or $M > (1-\lambda)^{-1}$ (assuming $\lambda > 0$, $\mu > 0$, and $\lambda + \mu < 1$).

We will prove the achievability parts of Theorem 2.1 in Section 3, and the converse parts in Section 4. The achievability results can actually be proved with Condition (2) in Definition 2.1 replaced by the stronger condition "$Q_{F_{a,m}}(\mathcal{D}_{a,m}) > 1 - \lambda$ for all m." However, it is *not* possible to similarly replace Condition (3) by "$Q_{F_{a,m}}(\mathcal{D}_{a'}) < \mu$ for all m and all $a' \neq a$" without affecting the results. Averaging over messages is essential in controlling the probability of a receiver wrongly deciding that it is the intended recipient.

3 Proofs of the achievability parts

The achievability proofs are based on those of [1]. The following two lemmas contain the basic ideas in the proofs.

Lemma 3.1 *Let P be an n-type on $\mathcal{X}$ for some $n \geq 1$ (i.e., P is a p.d. on $\mathcal{X}$ such that $nP(x)$ is an integer for all $x \in \mathcal{X}$), and let $d = |\mathcal{X}||\mathcal{Y}|$.*

(1) If $R' \geq 0$ and $J \leq \exp(nR')$, then there exist sequences $\mathbf{c}_1, \ldots, \mathbf{c}_J$ in $\mathcal{X}^n$, all of type P, and a partition of $\mathcal{Y}^n$ into subsets $\mathcal{C}_1, \ldots, \mathcal{C}_J$, such that $1 - W^n(\mathcal{C}_j|\mathbf{c}_j) \leq (n+1)^{4d} \exp[-nE'(R', P)]$ for all $j \in [J]$. Here, $E'(R', P)$ is a continuous function of (R', P) which is positive if $R' < I(P; W)$ and zero otherwise.

(2) If $R'' \geq 0$ and $K \leq \exp(nR'')$, then, for any $\mathbf{c} \in \mathcal{X}^n$ of type P and any $\mathcal{C} \subseteq \mathcal{Y}^n$, there exists a partition of $\mathcal{C}$ into subsets $\mathcal{C}_e, \mathcal{C}_1, \ldots, \mathcal{C}_K$, such that $W^n(\mathcal{C}_k|\mathbf{c}) = (1/K)[W^n(\mathcal{C}|\mathbf{c}) - W^n(\mathcal{C}_e|\mathbf{c})]$ for all $k \in [K]$, and $W^n(\mathcal{C}_e|\mathbf{c}) \leq (n+1)^d \exp[-nE''(R'', P)]$. Here, $E''(R'', P)$ is a continuous function of (R'', P) which is positive if $R'' < H(W|P)$ and zero otherwise.

Proof: Part (1) is a standard result in channel coding. See Theorem 5.2 on p. 165 of [3] for a proof. Part (2) can be proved by simple type-based arguments. For details, see [6]. □

Lemma 3.2 *Let $0 < \epsilon < 1$, and let $J \geq 2/\epsilon$, $S \geq 1$, and $N < \exp(\epsilon^2 S/2)$ be integers. Then, there exists an $N \times S$ array (N rows and S columns) with entries from $[J]$, any two rows of which are at a Hamming distance greater than $(1 - \epsilon)S$ from each other.*

Proof: See [6]. □

We will now briefly outline the achievability proofs. Details can be found in [6]. The first part of Lemma 3.1 implies the existence of an "equitype" transmission code (for the DMC W *without* feedback) of blocklength n with about $\exp[nI(P; W)]$ codewords — P is the common type of the codewords — whose maximal error probability over W decays exponentially with n. We will use this result as follows: given $R_1 < C$, the transmitter and receivers choose a type P such that $I(P; W) > R_1$, and agree on an equitype code as above. In the general case, the transmitter uses this code to reliably convey one of $M = \exp(nR_1)$ messages *and* an independently generated random variable of entropy about $n[I(P; W) - R_1]$ to the receivers. In the deterministic case, the transmitter simply conveys the message alone using the code.

By the second part of Lemma 3.1, the decoding set corresponding to any transmitted codeword can be partitioned into about $\exp[nH(W|P)]$ sets — all of which have *exactly* the same probability — and a remaining "error" set whose probability decays exponentially with n. Thus, by observing which of these subsets the channel output falls in, the transmitter and receivers can agree on a random variable of entropy about $nH(W|P)$. Since the codeword itself carries an entropy of about $nI(P; W)$ (in the general case) or nR_1 (in the deterministic case), the transmitter and receivers have a total *common randomness* of about $nI(P; W) + nH(W|P) = nH(PW)$ (general case) or $nR_1 + nH(W|P)$ (deterministic case).

Now suppose $R_2 < H(PW)$ (general case) or $R_2 < R_1 + H(W|P)$ (deterministic case), and $N = \exp[\exp(nR_2)]$. Then, by Lemma 3.2, we can find an $N \times S$ array whose rows are far apart in Hamming distance, with S roughly equal to $\exp[nH(PW)]$ or $\exp\{n[R_1 + H(W|P)]\}$ (in the respective cases). We can index the rows of the array by the N receivers. The common randomness gained by the codeword transmission can then be used by the transmitter and receivers to agree on a random column of this array (in either case). If the transmitter now sends the array element in that column and the row corresponding to the intended recipient (this can be done reliably with a constant number of channel uses), then the Hamming distance properties of the array enable the receivers to decide reliably whether or not the message is intended for them.

The idea of using channel noise to generate randomness for identification coding in the presence of feedback is one of the key principles in the results of [1]. The details of the idea there are somewhat different, and the method outlined here (based on the second part of Lemma 3.1) seems to allow a slightly simpler analysis. Lemma 3.2 is based on the arguments in Section III of [1], though it is not stated there in this form. It can in fact be used to prove all known achievability results in identification theory.

4 Proofs of the converse parts

Consider any sequence of $(n, N_n, M_n, \lambda, \mu)$ IT codes $\{(F_{a,m}, \mathcal{D}_{a,m})\}$ achieving the rate-pair (R_1, R_2), with $\lambda > 0$, $\mu > 0$, and $\lambda + \mu < 1$. We will now outline the ideas for bounding R_1 and R_2.

To begin with, $\{(F_{1,m}, \mathcal{D}_{1,m}) : m \in [M_n]\}$ is a sequence of (n, M_n) transmission codes with average error probability λ, for the DMC W with feedback. The encoding may involve randomization, but this does not help at all in a transmission code. Since $\lambda < 1$, Kemperman's strong converse to Shannon's theorem for DMCs with feedback [5] yields $\limsup n^{-1} \log M_n \leq C$, which implies that $R_1 = \liminf n^{-1} \log M_n \leq C$. However, it turns out that we can prove Kemperman's result here with very little additional effort, and will therefore not appeal to it directly. Our proof of this result is different from the original one.

The idea for bounding the identification rate is similar to that in [1]. Fix any $\gamma \in (\mu, 1 - \lambda)$. Suppose we could find subsets $\mathcal{D}_a^* \subseteq \mathcal{D}_a$ such that

$$(1/M_n) \sum_m Q_{F_{a,m}}(\mathcal{D}_a^*) > \gamma \quad \text{and} \quad |\mathcal{D}_a^*| \leq K_n, \quad \text{for each } a.$$

Then, since $(1/M_n) \sum_m Q_{F_{a,m}}(\mathcal{D}_{a'}^*) < \mu < \gamma$ for all $a' \neq a$, the sets $\mathcal{D}_a^*$ would have to be distinct. But this would imply that N_n is no bigger than the number of distinct subsets of $\mathcal{Y}^n$ whose size is at most K_n, which in turn is bounded above by $|\mathcal{Y}^n|^{K_n}$. Thus we would have

$$n^{-1} \log \log N_n \leq n^{-1} \log K_n + o(1).$$

The idea is to prove that if n is large enough then there exist subsets $\mathcal{D}_a^* \subseteq \mathcal{D}_a$ satisfying the above conditions, with $n^{-1} \log K_n$ being $\max_{P: I(P;W) \geq R_1 - \delta} H(PW) +$

$o(1)$ in the general case, and $n^{-1}\log M_n + \max_{P:I(P;W)\geq R_1-\delta} H(W|P) + o(1)$ in the deterministic case. Here, δ is an arbitrary positive number. The required bounds on the identification rate $R_2 = \liminf n^{-1}\log\log N_n$ in the two cases will then follow from the continuity in R of $\max_{I(P;W)\geq R} H(PW)$ and $\max_{I(P;W)\geq R} H(W|P)$.

To prove the above, we must first develop an appropriate notion of typical sequences. For any n-step strategy F, and $(x^n, y^n) \in \mathcal{X}^n \times \mathcal{Y}^n$, let $P_k^F(x|x^{k-1}, y^{k-1})$ be the probability that $X_k = x$ given $(X^{k-1}, Y^{k-1}) = (x^{k-1}, y^{k-1})$, under the strategy F. Let $P^F_{x^n,y^n}$ be the p.d. on $\mathcal{X}$ given by $P^F_{x^n,y^n}(x) = n^{-1}\sum_{k=1}^n P_k^F(x|x^{k-1}, y^{k-1})$.

The "typical set" $\mathcal{E}(F)$ for the strategy F is then defined as the set of (x^n, y^n) such that $W_F(x^n, y^n) > 0$, and

$$|N(x,y|x^n,y^n) - nP^F_{x^n,y^n}(x)W(y|x)| \leq n^{3/4}\sqrt{W(y|x)}$$

for all (x, y) (here, $N(x,y|x^n,y^n) = |\{k : (x_k, y_k) = (x,y)\}|$). The following lemma says that the typical set $\mathcal{E}(F)$ carries most of the probability under the p.d. W_F, and sequences in $\mathcal{E}(F)$ are "well-behaved" in various ways.

Lemma 4.1 *Let $d = |\mathcal{X}|\,|\mathcal{Y}|$. Then, for any n-step strategy F,*

$$1 - W_F(\mathcal{E}(F)) \leq dn^{-1/2}. \tag{1}$$

If $(x^n, y^n) \in \mathcal{E}(F)$, and y^n has type Q, then

(a) $\left|\log Q^n(y^n) + nH(P^F_{x^n,y^n}W)\right| \leq dn^{7/8}$.

(b) $\left|\log W^n(y^n|x^n) + nH(W|P^F_{x^n,y^n})\right| \leq dn^{7/8}$.

(c) $\left|\log\left(\frac{W^n(y^n|x^n)}{Q^n(y^n)}\right) - nI(P^F_{x^n,y^n}; W)\right| \leq 2dn^{7/8}$.

Proof: Essentially by Chebyshev's inequality. See [6] for details. □

For any $\alpha \geq 0$, let $\mathcal{B}_\alpha(F) = \{y^n : \exists\, x^n \in \mathcal{E}_{y^n}(F) \text{ such that } I(P^F_{x^n,y^n}; W) > \alpha\}$, where $\mathcal{E}_{y^n}(F) = \{x^n : (x^n, y^n) \in \mathcal{E}(F)\}$ is the "section" of $\mathcal{E}(F)$ at y^n.

Lemma 4.2 *Let $F_1, \ldots, F_M$ be n-step strategies, and $\mathcal{D}_1, \ldots, \mathcal{D}_M$ pairwise disjoint subsets of $\mathcal{Y}^n$. Let $\mathcal{B}^c_\alpha(F_m) = \mathcal{Y}^n - \mathcal{B}_\alpha(F_m)$. Then:*

(1) $(1/M)\sum_m Q_{F_m}(\mathcal{D}_m \cap \mathcal{B}^c_\alpha(F_m)) \leq (1/M)\exp\{n\alpha + o(n)\} + o(1)$.

(2) $|\bigcup_m \mathcal{D}_m \cap \mathcal{B}_\alpha(F_m)| \leq \exp\{n[\max_{P:I(P;W)\geq\alpha} H(PW)] + o(n)\}$.

(3) $|\bigcup_m \mathcal{D}_m \cap \mathcal{B}_\alpha(F_m)| \leq M\cdot\exp\{n[\max_{P:I(P;W)\geq\alpha} H(W|P)] + o(n)\}$ *if $F_1, \ldots, F_M$ are deterministic.*

Proof: Appendix. □

We will now return to the sequence of $(n, N_n, M_n, \lambda, \mu)$ IT codes $\{(F_{a,m}, \mathcal{D}_{a,m})\}$ at the beginning of this section, and complete the proofs of the converses. First, note that if $\alpha = C$ then $\mathcal{B}_\alpha(F_{1,m})$ is empty for all m, so that $\mathcal{D}_{1,m} = \mathcal{D}_{1,m} \cap \mathcal{B}^c_\alpha(F_{1,m})$. Therefore, $1 - \lambda < (1/M_n)\sum_m Q_{F_{1,m}}(\mathcal{D}_{1,m}) \leq (1/M_n)\exp[nC + o(n)] + o(1)$, where the second inequality is by Part (1) of Lemma 4.2, applied with $\alpha = C$. Since $\lambda < 1$, the above inequalities imply that $\limsup n^{-1}\log M_n \leq C$, whence $R_1 \leq C$. We have now proved Kemperman's strong converse for DMCs with feedback, and the

required bound on the transmission rate for general IT codes as well as deterministic ones.

To bound the identification rate, we will choose the set $\mathcal{D}_a^*$ to be $\bigcup_m \mathcal{D}_{a,m} \cap \mathcal{B}_\alpha(F_{a,m})$ for each a, with $\alpha = R_1 - \delta$ (δ is an arbitrary positive number). This amounts to throwing away all sequences y^n in the decoding region $\mathcal{D}_{a,m}$ which are, roughly speaking, either "atypical" (i.e., $\mathcal{E}_{y^n}(F_{a,m})$ is empty) or have "low mutual information" (i.e., $I(P_{x^n,y^n}^{F_{a,m}}; W) \leq R_1 - \delta$ for all $x^n \in \mathcal{E}_{y^n}(F_{a,m})$). The intuition is that, if n is large, such sequences cannot contribute significantly to the probability of $\mathcal{D}_a$ because the IT code is required to transmit messages at rate R_1. More precisely, it can be proved that, for any a,

$$(1/M_n) \sum_m Q_{F_{a,m}}(\mathcal{D}_a^*) > 1 - \lambda - (1/M_n) \exp[n(R_1 - \delta) + o(n)] - o(1),$$

which exceeds γ for all large n, as required earlier. In the inequality above, we have once again used Part (1) of Lemma 4.2, this time with $\alpha = R_1 - \delta$.

The removal of the "inessential" sequences also trims down $\mathcal{D}_a$ to the right size: $|\mathcal{D}_a^*|$ is bounded above by the right-hand sides of the inequalities in Parts (2) and (3) of Lemma 4.2 in the general and deterministic cases, respectively, with α replaced by $R_1 - \delta$ and M by M_n. Thus, the sets $\mathcal{D}_a^*$ have all the properties postulated earlier, and the converses are proved.

References

[1] R. Ahlswede and G. Dueck. Identification in the presence of feedback - a discovery of new capacity formulas. *IEEE Transactions on Information Theory*, Vol. 35(No. 1), January 1989.

[2] R. Ahlswede and G. Dueck. Identification via channels. *IEEE Transactions on Information Theory*, Vol. 35(No. 1), January 1989.

[3] I. Csiszar and J. Korner. *Information Theory: Coding Theorems for Discrete Memoryless Systems.* Academic Press, 1981.

[4] T.S. Han and S. Verdú. New results in the theory of identification via channels. *IEEE Transactions on Information Theory*, Vol. 38(No. 1), January 1992.

[5] J.H.B. Kemperman. Strong converses for a general memoryless channel with feedback. *Trans. 6th Prague Conf. Information Theory, Stat. Dec. Fct's and Rand. Proc.*, 1973.

[6] S. Venkatesan and V. Anantharam. Identification plus transmission over channels with perfect feedback. Technical Report UCB/ERL M96/34, Electronics Research Laboratory, Univ. of California, Berkeley, May 1996.

Embedded Codes for Combined Source-Channel Coding*

Shrinivas Gadkari and Kenneth Rose
Center for Information Processing Research
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106, USA

Abstract

Motivated by the problem of coding a source over a mobile communication channel, we address the issues related to design and performance analysis of unequal error protection (UEP) schemes, for operation over time-varying Gaussian channels. UEP schemes fall into two categories (1) Non-embedded codes and (2) Embedded codes. To analyze the relative merits of these two classes we first develop a general framework for analysis of practical UEP schemes. Subsequently we demonstrate that, depending on the desired difference in the protection levels, one scheme outperforms the other. The analysis provides insights into the factors that dictate the choice of one scheme over the other.

1 Introduction

In this paper we address the problem of coding a source over a time varying Gaussian channel. We focus our attention on this channel model due to its applicability to mobile communication systems. A primary concern for these systems is the multipath fading of the received signal. Due to surrounding obstacles, the received signal is a sum of many reflected replicas of the transmitted signal attenuated and delayed by different amounts. This causes the net received signal to have a random phase and amplitude. Moreover, as the receiver moves, the strength of the received signal varies resulting in time-varying fading. Often, fading is a rapidly time varying phenomenon and hence it is not possible to keep the transmitter informed about the instantaneous value of the received signal power. The receiver can, however, keep track of the time-variations in the received signal power. Hence the scenario that we encounter is that of an informed receiver and an uninformed transmitter [3]. A naive approach to transmission over such a channel is to perform the code design for the worst case channel conditions. This, however, restricts the received signal to have the same quality irrespective of channel conditions. In fact this approach is sub optimal for all but the maxmin performance measure. An attractive alternative to this scheme is to

*This work was supported in part by National Science Foundation under grant no. NCR-9314335, University of California MICRO program, Cisco Systems, DSP Group, Inc., DSP Software Engineering, Echo Speech Corp., General Electric, Hughes Electronics, Intel, Moseley Associates, Qualcomm, Inc., Rockwell International Corp., Speech Technology Labs, and Texas Instruments, Inc.

use a successive refinement (or multi-stage) source coding scheme. This is followed by channel coding that provides unequal error protection to the various stages of the source coder. Specifically, the coarser stages are given more protection, compared to the fine detail stages. The receiver uses the information about the noise level in the channel[1] to decode as many stages of the encoded source as can be done reliably. Hence, at the receiver we have a graceful degradation of the received signal. When the channel is clean, all the stages can be successfully decoded producing the best quality of the decoded signal. As the level of channel noise increases we have a corresponding drop in the quality due to inability of the receiver to reliably decode the fine detail stages. System design for such a channel involves the following two issues. (1) Design of an appropriate multi-stage source coder. (2) Design of an unequal error protection (UEP) scheme for this source coder. Previous works that have addressed system design for such a channel include [2, 4]. In this work we address the issue of designing UEP scheme to protect the various stages of the multi-stage source coder.

In his classic paper on broadcast channels[1], Cover showed that embedded or superimposed codes always outperform time-sharing/ non-embedded coding in applications that require unequal error protection. In the present work, however, we demonstrate that for practical UEP schemes, Cover's result does not hold in general. The contributions of this work can be summarized as follows. We develop a framework for performance evaluation of practical UEP codes. Using this framework we first show that, at times non-embedded codes outperform embedded codes. Consequently we provide insight into the conditions under which one scheme outperforms the other.

2 Problem Statement

We assume a pre-designed mulatf-stage source coder. Our interest is in providing appropriate unequal error protection to the various stages of the source coder. We focus our attention on two stages of this coder. We denote them at stages 1 and 2, where the more significant stage is termed 1. The bits generated by these two stages will be represented by R_1 and R_2, respectively. We desire to provide error protection to these two stages such that R_1 bits can be decoded successfully as long as the variance of tha Gaussian noise on the channel is less than N_1, while the R_2 bits can be decoded successfully when the channel noise has variance smaller than N_2, where $N_1 > N_2$. The available resources for use are: M modulation intervals, and transmission energy. Our goal is to minimize the total transmission energy required to maintain the desired level of protection.

3 Non-embedded Codes

We first describe and analyze a class of UEP schemes that will be referred to as non-embedded codes, to be later contrasted with the class of embedded coding schemes. The scheme can be generally formulated as: Use M_1 modulation intervals to transmit

[1]Although it is the received signal power that varies, while the noise power remains constant, we can easily map the problem to an equivalent one where the received signal power is constant, while the noise power varies.

R_1 bits using transmission energy E_1. On the remaining $M_2 = M - M_1$ modulation intervals we transmit the R_2 bits using an amount of transmission energy E_2. The problem that needs to be solved is to find the optimal set of values $\{(M_1, E_1), (M_2, E_2)\}$ that obtains the desired level of protection while minimizing $E_{tot} = E_1 + E_2$. To simplify the discussion we introduce the following notation where all the quantities are normalized by M. Let $M_i = \alpha_i M$ (where $\alpha_1 + \alpha_2 = 1$), $R_i = r_i M$, $E_i = e_i M$ and $E_{tot} = e_{tot} M$. The quantities r_i represent the bits transmitted per modulation interval and measure the spectral efficiency of the channel coding schemes.
We can identify two special cases of the general scheme as.
Transmission Energy Allocation (TEA) : Here we restrict $\alpha_i = \frac{r_i}{r_1 + r_2}$, while optimize the values of e_1 and e_2, i.e. we use the same channel code for both the stages by time-sharing the available modulation intervals proportional to the respective bit rates. However, the power of the channel code is changed by varying the transmission energy per modulation interval allocated to each stage, yielding different levels of protection to the two bit streams. See [10, 9] for some practical schemes based on TEA.
Code Optimization (CO) : Here we use $e_1 = e_2 = e_{tot}$, and optimize the values of α_1, α_2. In other words, we keep the energy transmitted during each modulation interval constant, while allocating different modulation intervals for transmission of the two bit streams. Many speech coding schemes implement CO scheme by employing different error correcting codes to protect different classes of bits, e.g. [7, 8].
We will focus our attention largely on the most general scheme wherein, both (α_1, α_2) and (e_1, e_2) are optimized. We will refer to this scheme as TEACO. However, for the sake of completeness we will briefly comment on the performance of TEA and CO relative to TEACO.

3.1 Framework for Analysis of UEP Codes

In this section we develop a framework for performance analysis of practical codes. The framework will be used throughout this paper to analyze the performance of UEP codes.

We begin by introducing the following notation to compactly describe a channel code: A (r, e) code, C, transmits r bits per modulation interval using an amount of transmission energy e per modulation interval. Let N^* be the maximum variance of the Gaussian noise per modulation interval that can be tolerated by the code, while maintaining almost error-free transmission. If C were an ideal code - a code that achieves capacity - then the quantities r, e and N^* would be related by Shannon's capacity formula for Gaussian channels :

$$r = \frac{1}{2} \log_2 \left(1 + \frac{e}{N^*}\right). \tag{1}$$

We introduce a correction parameter λ to account for the fact that the code is not ideal. λ can be interpreted as follows : $10 \log_{10} \lambda$ is the increase in dB of the SNR over the channel required for almost noise-free transmission (relative to the SNR required by an ideal code of same rate.) We rewrite (1) for a practical (r, e) channel code, where almost error-free transmission is achieved as long as the variance of the

Rate of Code	Constraint Length	λ
1/2	7	2.6
1/2	6	2.8
1/2	5	3.0
1/2	4	3.4
1/2	3	3.6
1/2	2	3.9
1/3	7	2.7
1/3	5	3.2
1/3	3	3.9

Table 1: Here we list the values of SNR loss factor λ for some binary convolutional codes with binary modulation. We can see that as we increase the complexity (quantified by the constraint length), the value of λ decreases.

Gaussian channel noise is less than N, where r, e and N are related by

$$r = \frac{1}{2}\log_2\left(1 + \frac{e}{N\,\lambda}\right). \tag{2}$$

We call N in (2) the maximum noise immunity (MNI) of the code. The SNR loss factor, λ, depends on the complexity of the channel code and has to be estimated empirically. To illustrate the relation of λ with complexity, we list its values for some practical binary codes with binary modulation in Table 1. We see that, using convolutional codes with 64 states (constraint length = 7) a value of λ in the range of 2.6 can be obtained. For two dimensional uncoded QAM constellations the value of λ is about 8. Trellis coded modulation can achieve coding gains in the range of about $3 - 4.5$ dB, with 3 dB corresponding to relatively simple 4/8 state codes, while the more complex trellis codes with 128/256 states yield about 4.5 dB coding gains, e.g. [5, 6]. Trellis codes with sequential decoding can yield upto 5.4 dB coding gains[6]. The corresponding values of λ for coded modulation systems employing QAM modulation are in the range of 4 to 2.3.

3.2 Performance of Non-Embedded Codes

In this section we will consider the problem of optimal allocation of resources for the general TEACO problem. We will assume that both the codes have the same value for λ. We will also restrict the analysis to the case $r_1 = r_2 = r$, though the analysis can be easily extended to the case $r_1 \neq r_2$. The prime reason to emphasize the case $r_1 = r_2$ stems from the fact that in many successive approximation based quantizers, such as MSVQ, all the stages often use equal bits. We recognize that the quantities N_1 and N_2 described in the problem statement are in fact the desired MNIs for the two stages. To meet these specifications, we require

$$r = \frac{\alpha_i}{2}\log_2(1 + \frac{e_i}{\lambda\,N_i}). \tag{3}$$

where, $\alpha_1 + \alpha_2 = 1$. We need to choose α_i such that the total energy $e_{tot} = \alpha_1\,e_1 + \alpha_2\,e_2$ is minimized.

Before presenting the complete result of this optimization, we mention an important property of the optimal solution.

Theorem 1 : If $N_1 > N_2$, the optimal solution satisfies $\alpha_1 > \alpha_2$. Hence, the more significant data is protected by a lower rate code.

Proof : We have

$$e_{tot} = \lambda \left[\alpha_1 N_1 (2^{\frac{2r}{\alpha_1}} - 1) + \alpha_2 N_2 (2^{\frac{2r}{\alpha_2}} - 1) \right]. \tag{4}$$

It is easy to show that $\alpha(2^{\frac{2r}{\alpha}} - 1)$ is monotonic decreasing in α. Hence, if we choose $\alpha_2 > \alpha_1$, e_{tot} can be decreased by interchanging the values of α_1 and α_2.

The complete optimization result can be summarized as:

Result 1 : The parameters α_1 and α_2 are completely determined by the ratio of MNIs - $\frac{N_1}{N_2}$, and r as

$$\frac{N_1}{N_2} = \frac{2^{\frac{2r}{\alpha_2}}(1 - \frac{2r\,ln2}{\alpha_2}) - 1}{2^{\frac{2r}{\alpha_1}}(1 - \frac{2r\,ln2}{\alpha_1}) - 1}. \tag{5}$$

The dependence of (α_1, α_2) on the MNIs only through the ratio $\frac{N_1}{N_2}$ will be used later when we compare the performance of the embedded codes against that of the TEACO codes. Using (5) we obtain the values of α_i, which can be used in (3 to obtain the values of e_i. It is interesting to note that (5) is independent of λ.

To illustrate the relative performance of TEACO, consider Figure 1 where, we plot the the values of (N_1, N_2) that can be obtained by TEACO, TEA and CO for a fixed value of e_{tot}. It can be seen that TEACO achieves a significantly better performance as compared to TEA. It is also interesting to observe that the performance of CO is close to that achieved by TEACO. However, to achieve this performance we need to select the right rate code. In practice, we might be constrained to choose from a small collection of available channel codes. Under these circumstances, using only CO, severely limits the amount of unequal error protection that can be provided to the two stages. However, TEACO is able to achieve close to optimal performance even with a few number of codes to choose from. This point is illustrated in Figure 2, where, we restrict the codes to be among 5 possible codes. It can be seen that TEACO method has a clear advantage over CO. Henceforth, when we refer to non-embedded coding, we will mean TEACO.

4 Embedded Codes

Embedded codes differ for the TEACO codes described in the last section in the following respect. Instead of transmitting the two stages of the source coder on separate modulation intervals, we use all the M available modulation intervals to transmit all the bits. This is achieved as follows. We design two codes C'_1 and C'_2 with parameters (r, e'_1) and (r, e'_2). We select codewords U_1 and U_2 for these two codes and transmit the sum $U = U_1 + U_2$ on the channel. At the receiving end we use the received vector $\tilde{U} = U + noise$ and decode $\hat{U}_1$. We subsequently subtract $\hat{U}_1$ from $\tilde{U}$ and decode $\hat{U}_2$ from the difference $\tilde{U} - \hat{U}_1$. Before we proceed with the performance analysis of embedded codes we introduce the following terminology: we will refer to codes C'_1 and C'_2 as outer and inner codes respectively.

4.1 Analysis of Performance

The framework developed earlier will now be used to analyze the performance of embedded codes. We will concentrate our attention largely on the case when both the inner and the outer codes have the same SNR loss factor, λ. For the inner code C_2', to meet the specifications on MNIs we require

$$r = \frac{1}{2}\log_2\left(1 + \frac{e_2'}{N_2\,\lambda}\right).$$

For outer code C_1', the codeword of the code C_2' acts as an additional noise (in addition to the channel noise) Thus to meet the given specifications the outer code should be able to withstand the combined effect of channel noise of variance N_1 and the "noise" due to the inner code C_2'. Approximating the combined effect of "noise" as a Gaussian noise of variance $N_1 + e_2'$,[2] we need to design the outer code with MNI of $N_1 + e_2'$. Hence we have,

$$r = \frac{1}{2}\log_2\left(1 + \frac{e_1'}{[N_1 + e_2']\,\lambda}\right).$$

If we denote the total energy required per modulation interval as e'_{tot}, we can relate it to r, N_1, N_2 as

$$e'_{tot} = \theta(N_1 + N_2) + \theta^2 N_2, \tag{6}$$

where, $\theta = \lambda(2^{2r} - 1)$. We give the N_1 versus N_2 plots for embedded codes and TEACO codes if Figure 3. We observe that *contrary to the ideal result in [1] at times, practical embedded codes perform poorly compared to non-embedded TEACO codes.*

The results of Figure 3, motivate the following important question : If we desire to achieve the specified noise immunity for the two stages of the source coder, under what conditions does embedded coding scheme obtain energy savings over the other method. This question is answered in the following theorem.

Theorem 2 : For codes with the same value of SNR loss factor λ, the quantity $\frac{e'_{tot}}{e_{tot}}$ depends on the prescribed MNIs only through the ratio $\frac{N_1}{N_2}$.

Proof : The average energy per modulation interval required by the TEACO, e_{tot}, is given by (4). Hence, we have

$$\frac{e'_{tot}}{e_{tot}} = \frac{\theta(\frac{N_1}{N_2} + 1) + \theta^2}{\frac{N_1}{N_2}\alpha_1\lambda(2^{\frac{2r}{\alpha_1}} - 1) + \alpha_2\lambda(2^{\frac{2r}{\alpha_2}} - 1)}. \tag{7}$$

Next consider (5) which shows that the choice of (α_1, α_2) depends on the ratio $\frac{N_1}{N_2}$. Hence the result follows.

To compare the performance of embedded codes against TEACO, we present the plots of $\frac{e'_{tot}}{e_{tot}}$ versus $\frac{N_1}{N_2}$ in Figures 4a and 4b. We make the following important observations :

(1) At low values of $\frac{N_1}{N_2}$, TEACO is substantially outperforms the embedded coding

[2]This approximation is asymptotically exact, and is widely used in discussion of broadcast channels [1]. In our preliminary experiments we found that this approximation is in fact conservative, i.e. the combined effect of the inner code and the noise, often produces less degrading effect on the performance of the outer code than, what is produced by a Gaussian noise of variance $N_1 + e_2'$.

scheme. The gains of embedded coding start showing when we are interested in a large ratio of $\frac{N_1}{N_2}$. This is typically the situation when we are providing protection to stages of a multi-stage source coder that code signals of significantly different importance. For providing UEP to stages of nearly the same importance, TEACO is certainly preferable.
(2) The gains increase with r. In fact at high spectral efficiencies, embedded coding appears to yield impressive savings of the order of several dB over TEACO. Surprisingly, the ratio $\frac{N_1}{N_2}$ at which we start gaining over TEACO appears to vary very little with r.

4.2 Complexity Tradeoffs

Until now, our discussion was restricted to the case when both the inner and outer codes have the same values of SNR loss factor λ. Here, we address some issues related to the case when the inner and the outer codes have different values of λ_1 and λ_2. This case is motivated by the following important practical consideration. We have already seen that the value of λ is related to the complexity of the code. Hence, a smaller value of λ can be obtained by investing more complexity in the channel code. The question we pose is: Would it be better to invest complexity in the inner code or the outer code. The following theorem answers this question.
Theorem 3: If $N_1 > N_2$ then the optimal solution satisfies $\lambda_1 \leq \lambda_2$.
Proof : Let λ_1 and λ_2 be the SNR loss factors for the other code and the inner code respectively. We can write e'_{tot} as

$$e'_{tot} = (2^{2r} - 1)\left(\lambda_1 N_1 + \lambda_2 N_2 + (2^{2r} - 1)\lambda_1 \lambda_2 N_2\right). \tag{8}$$

Since $N_1 > N_2$, we have $\lambda_1 \leq \lambda_2$, else we can reduce e'_{tot} by switching the codes.
Thus in embedded coding, it pays to invest additional complexity in the outer code.

5 Conclusion and Discussions

In this paper we addressed the issue of design of unequal error protection (UEP) codes for operation over a time-varying Gaussian channel. We considered two classes of codes, viz. Non-embedded codes and embedded codes. We developed a general framework for analysis of UEP codes and used it to demonstrate the fact that practical embedded codes can be inferior in performance compared to non-embedded codes. We discussed the conditions when one scheme outperforms the other. It was shown that embedded codes are preferable when the two bit streams under consideration need protection against noise levels differing by many dB. It was also demonstrated that the gains are more pronounced when we are interested in high spectral efficiency. We addressed issues relating to complexity tradeoff between inner and outer codes in an embedded coding system.

Current work involves a more detailed study of the issues like maximum achievable performance gains, critical ratio $\frac{N_1}{N_2}$ at which embedded codes outperform TEACO, extensions to the general case $r_1 \neq r_2$, etc.. The counterpart of this problem is that of design of an appropriate source coder to generate the bits streams that need unequal error protection. Preliminary ideas to tackle this problem can be found in [11].

References

[1] T. Cover, "Broadcast Channels", *IEEE Trans. on Infrom. Theory*, vol. IT-18, pp. 2-14, Jan. 1972.

[2] K. Ramchandran, A. Ortega, K. M. Uz, and M. Vetterli, "Multiresolution broadcast for HDTV using combined source-channel coding," *IEEE J. on Sel. Areas on Comm.*, vol. 11, pp. 6-23, Jan 1993.

[3] M. Khansari and M. Vetterli, "Time-varying channels with side information and separation principle",Electronics Research Laboratory Rept., UC Berkeley, Memorandum No. UCB/ERL M94/103.

[4] I. Kozintsev and K. Ramchandran, "Multiresolution Joint Source-Channel Coding using Embedded Constellations for Power Constrained Time Varying Channels", *ICASSP-96*, pp. 2343-2346.

[5] G. Ungerboeck, "Trellis coded modulation with redundant signal sets, Part I and II", *IEEE Commun. Mag.* vol. 25, pp. 5-22, Feb. 1987.

[6] F. Wang and D. Costello, "Probabilistic construction of large constraint length trellis codes for sequential decoding", *IEEE Trans. on Commun.* vol. 43, pp. 2439-48, Sept. 1995.

[7] H. Nielsen, K. B. Mihhelsen, et. al., "Comparative study of error correction coding schemes for the GSM half-rate channel", *ICASSP-92*, vol. 2,pp. 129-32.

[8] R. A. Salami, et. al., "Performance of error protected binary pulse excitation coders at 11.4 kb/s over mobile radio channels", *ICASSP-90*, pp. 473-76.

[9] P. Hoeher, "Unequal Error Protection for Digital Mobile DA-CDMA Radio Systems", *Globecom-94*, pp. 1236-41.

[10] S. Gadkari and K. Rose, "Robust vector quantisation by transmission energy allocation", *Electronics Letters*, vol. 32, pp. 1451-3, 1 Aug. 1996.

[11] S. Gadkari and K. Rose, "Transmission Energy Allocation for Robust Multi-Stage Vector Quantization", Proc. of Allerton Conf. on communication control and computing, 1996.

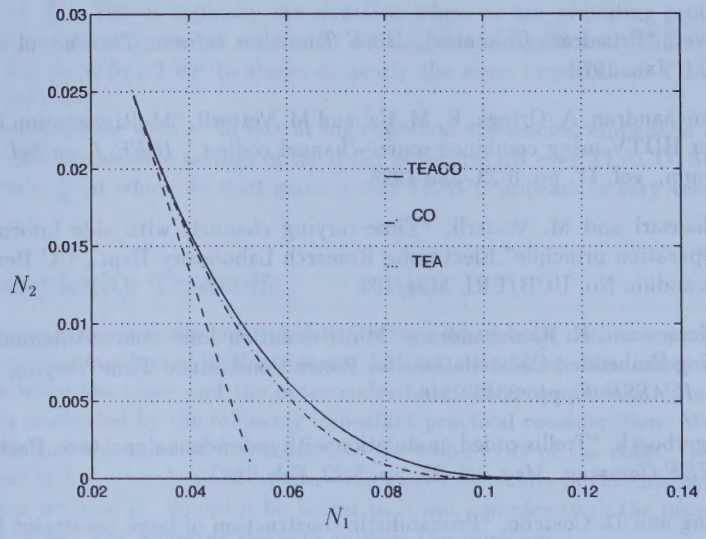


Figure 1: Comparison of N_1 v/s N_2 values that can achieved by TEA,CO and TEACO for same value of e_{tot}. The SNR loss factor $\lambda = 2.5$ and $r = 1$ were used.

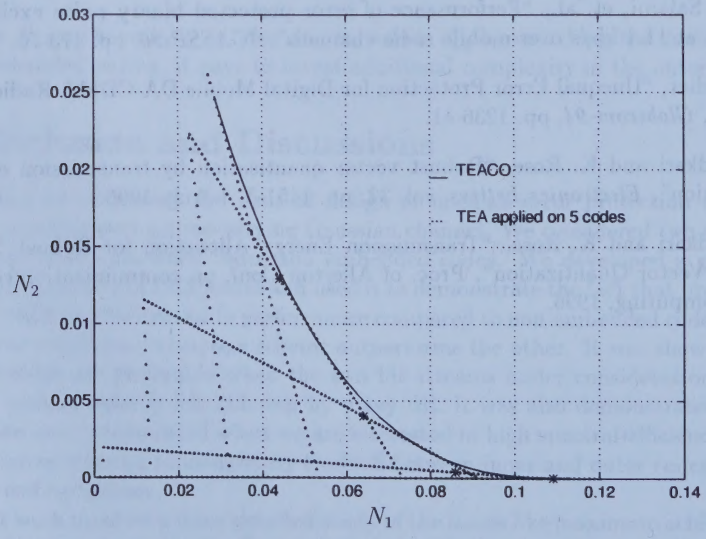


Figure 2: Here the ideal TEACO curve is approximated by performing TEA on 5 codes ($\lambda = 2.5$ and $r = 1$).

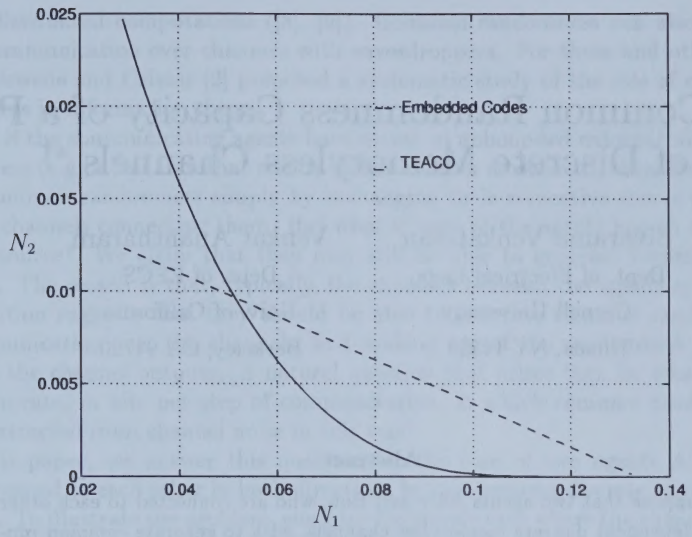
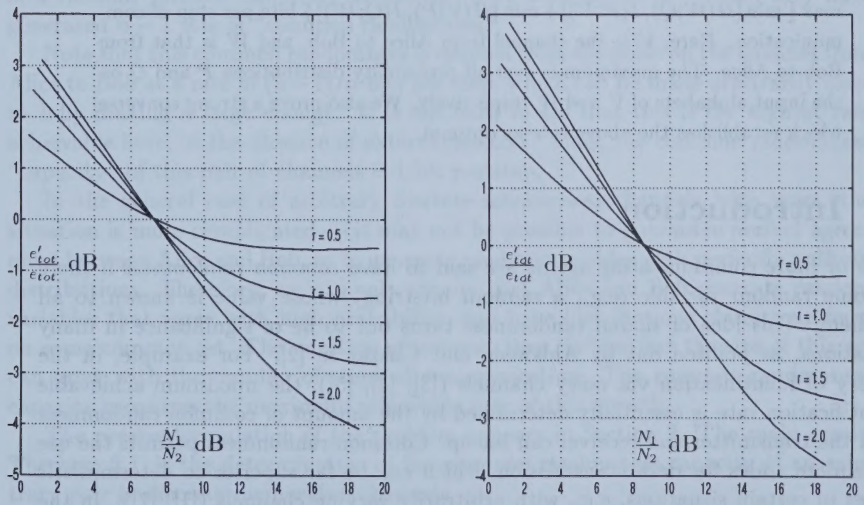


Figure 3: Comparison of N_1 v/s N_2 values that can achieved by TEACO and embedded codes, for the same value of e_{tot} and e'_{tot} ($\lambda = 2.5$ and $r = 1$).



(a) $\lambda = 2.5$ (b) $\lambda = 3.0$

Figure 4: Here the ratio $\frac{e'_{tot}}{e_{tot}}$ is plotted against $\frac{N_1}{N_2}$ for various values of r. We see that the value of $\frac{N_1}{N_2}$ at which embedded coding starts gaining over TEACO varies with λ but the variation with r is very small. Also note the increasing gains at higher spectral efficiencies.

The Common Randomness Capacity of a Pair of Discrete Memoryless Channels *†

Sivarama Venkatesan
Dept. of Electrical Engg.
Cornell University
Ithaca, NY 14853

Venkat Anantharam
Dept. of EECS
Univ. of California
Berkeley, CA 94720

Abstract

Suppose that two agents Alice and Bob, who are connected to each other by independent discrete memoryless channels, wish to generate *common randomness*, i.e., they want to agree on a common random number. We show that, even if Alice and Bob do not have access to any external random number generators, they can communicate interactively over the two channels, and exploit *channel noise* to generate common randomness at a rate of $\max\{\min[H(W|Q), I(P;V)] + \min[H(V|P), I(Q;W)]\}$ bits per step of communication. Here, V is the channel from Alice to Bob, and W is that from Bob to Alice. The maximum is over all probability distributions P and Q on the input alphabets of V and W respectively. We also prove a strong converse which establishes the above rate as optimal.

1 Introduction

Two or more communicating agents are said to have *common randomness* if there is some random variable (e.g., a random bitstring) whose value is known to all of them. This idea of shared randomness turns out to be of significance in many situations, as pointed out by Ahlswede and Csiszar in [2]. For example, in the theory of identification via noisy channels ([3], [4], [5]), the maximum achievable identification rate is essentially determined by the amount of common randomness that the transmitter and receiver can set up. Common randomness permits the use of random codes for data transmission, which can be far superior to deterministic codes in certain situations, e.g., with arbitrarily varying channels ([1], [7]). In the theory of communication complexity, it is known that common randomness can significantly reduce the amount of inter-processor communication required to perform

*Research supported by NSF IRI 9005849, IRI 9310670, NCR 9422513, and the AT&T Foundation.

†Address all correspondence to the first author (currently at U.C. Berkeley as an Exchange Scholar): 341, Cory Hall, Dept. of EECS, Univ. of California, Berkeley, CA 94720.

certain distributed computations ([8], [9]). Common randomness can also enable secure communication over channels with eavesdroppers. For these and other reasons, Ahlswede and Csiszar [2] proposed a systematic study of the role of common randomness in information theory.

Now, if the communicating agents have access to unbounded external sources of randomness (e.g., a fair coin that can be tossed as often as desired), then they could set up common randomness simply by exchanging their respective source outputs over the channels connecting them. But what if none of the agents has an external random source? We claim that they may still be able to generate common randomness. The reason is that, typically, the channels between the agents are noisy, and intuition suggests that they should be able to generate common randomness by communicating over the channels, and making use of the randomness (due to noise) in the channel outputs. A natural question that arises then is: what is the maximum rate, in bits per step of communication, at which common randomness can be extracted from channel noise in this way?

In this paper, we answer this question for the case of two agents Alice and Bob connected to each other in both directions by independent discrete memoryless channels. To illustrate the problem, consider the simple case where the channel from Alice to Bob is binary symmetric with crossover probability $1/2$, and that from Bob to Alice is a noiseless binary channel. Suppose Alice transmits 0 in n successive steps to Bob, and in step k, $2 \leq k \leq n$, Bob echoes the random output he received in step $k-1$ back to Alice. Then, after n steps, both Alice and Bob know the value of a random variable uniformly distributed over a set of size 2^{n-1}, i.e., they have generated $n-1$ bits of "common randomness."

Note that this common randomness is derived from the noise on the channel from Alice to Bob at a rate of $(n-1)/n$ bits per step, which can be made arbitrarily close to 1 by making n large enough. It is not hard to see that this is the highest rate achievable here, in the absence of external sources. Thus, the common randomness "capacity" of this pair of channels is 1 bit per step.

In the general case of arbitrary discrete memoryless channels both ways, the situation is more complicated — it may not be possible to guarantee perfect agreement between Alice and Bob, or to generate random variables with perfectly uniform distributions. Therefore, we will only require that Alice and Bob generate random variables that agree with high probability, and have distributions close to uniform on some common set. The question of interest, then, is how fast the size of this set can grow with the number of steps of communication. The common randomness capacity measures the maximum achievable rate of this growth.

The precise formulation of the problem appears in Section 2. The main result, Theorem 2.1, is the determination of the common randomness capacity of an arbitrary pair of discrete memoryless channels.

In the rest of the paper, a discrete memoryless channel (DMC) with input alphabet $\mathcal{Z}$, output alphabet $\hat{\mathcal{Z}}$, and transition probability function U will be denoted by $(\mathcal{Z}, \hat{\mathcal{Z}}, U)$, or just U if no confusion can result. $\mathcal{P}(\mathcal{Z})$ will denote the set of all probability distributions on the set $\mathcal{Z}$, and $\mathcal{P}_n(\mathcal{Z}) \subseteq \mathcal{P}(\mathcal{Z})$ will denote the set of all n-types, i.e., $P \in \mathcal{P}_n(\mathcal{Z})$ iff $nP(z)$ is an integer for all $z \in \mathcal{Z}$. $\mathcal{W}(\mathcal{Z}, \hat{\mathcal{Z}})$ will denote the set of all DMCs with input alphabet $\mathcal{Z}$ and output alphabet $\hat{\mathcal{Z}}$. If L is a positive

integer, then $[L]$ will denote the set $\{1, 2, \ldots, L\}$. All logarithms and exponentials will be to the base two.

2 Statement of problem and result

Let the DMC from Alice to Bob be $(\mathcal{X}, \hat{\mathcal{X}}, V)$, and that from Bob to Alice be $(\mathcal{Y}, \hat{\mathcal{Y}}, W)$. To generate common randomness, Alice and Bob communicate interactively with each other for a certain number of steps. In each step, Alice transmits a symbol to Bob across V and, simultaneously, Bob transmits a symbol to Alice across W. These symbols are determined by an agreed-upon strategy, as functions only of all the past receptions available to the respective senders.

Formally, an *n-step strategy* is a pair (f, g), with $f = (f_1, f_2, \ldots, f_n)$ and $g = (g_1, g_2, \ldots, g_n)$. Here, $f_k : \hat{\mathcal{Y}}^{k-1} \to \mathcal{X}$ and $g_k : \hat{\mathcal{X}}^{k-1} \to \mathcal{Y}$. Under this strategy, Alice sends $X_k = f_k(\hat{Y}^{k-1})$ and Bob sends $Y_k = g_k(\hat{X}^{k-1})$ in step k, $1 \leq k \leq n$, where $\hat{X}^{k-1}$ and $\hat{Y}^{k-1}$ are the sequences of outputs received by Bob and Alice, respectively, in the previous $k-1$ steps. The probability that $(\hat{X}^n, \hat{Y}^n) = (\hat{x}^n, \hat{y}^n)$ is then $\prod_{k=1}^{n}[V(\hat{x}_k|f_k(\hat{y}^{k-1})) \cdot W(\hat{y}_k|g_k(\hat{x}^{k-1}))]$.

After n steps, each agent separately decides whether the attempt to generate common randomness was successful or not, and in the former case computes a random output. Alice's decision is based only on $\hat{Y}^n$, and Bob's is based only on $\hat{X}^n$. Their random outputs take values in the common finite set $[K] = \{1, \ldots, K\}$. Formally, Alice computes $S = S(\hat{Y}^n)$ and Bob computes $T = T(\hat{X}^n)$, where $S : \hat{\mathcal{Y}}^n \to \{e\} \cup [K]$ and $T : \hat{\mathcal{X}}^n \to \{e\} \cup [K]$. Here, e is a symbol indicating failure to generate common randomness.

The quadruple (f, g, S, T) defines an (n, K) *protocol* for generating common randomness. Of course, the "amount" of randomness generated by the protocol, and the extent to which it is "common," are determined by the joint distribution of S and T. Ideally, we would like $Pr\{S(\hat{Y}^n) = T(\hat{X}^n) = l\} = 1/K$ for each $l \in [K]$, with K as large as possible. Then, S and T would be equal with probability 1, and uniformly distributed over $[K]$ (there would be no "failure" events of positive probability).

In general, it is not possible to satisfy this requirement except in the trivial case $K = 1$. Therefore, we will have to settle for *approximate* equality and uniformity of S and T. To this end, we define (f, g, S, T) to be an (n, K, λ) protocol if

$$\frac{1-\lambda}{K} \leq Pr\{S(\hat{Y}^n) = T(\hat{X}^n) = l\} \leq \frac{1+\lambda}{K} \quad \text{for each } l \in [K]. \tag{1}$$

Then, if λ is a small positive number, each agent generates a random output with distribution close to uniform on $[K]$, and these outputs agree with high probability. The probability that either agent declares failure to generate common randomness is also small. Note that if $\lambda \geq 1$ we can satisfy (1) with arbitrarily large K, for any n, simply by taking $S(\hat{y}^n)$ and $T(\hat{x}^n)$ to be e for all $(\hat{x}^n, \hat{y}^n)$.

Define $K(n, \lambda)$ to be the largest K such that there exists an (n, K, λ) protocol. The main result proved here is the following:

Theorem 2.1 (Main Theorem) *Let*

$$R^*(V,W) = \max_{\substack{P\in\mathcal{P}(\mathcal{X}) \\ Q\in\mathcal{P}(\mathcal{Y})}} \Big\{ \min\left[H(W|Q), I(P;V)\right] + \min\left[H(V|P), I(Q;W)\right] \Big\}. \qquad (2)$$

Then:

(a) (Direct part) $\liminf_{n\to\infty}(1/n)\log K(n,\lambda) \geq R^*(V,W)$ *for all* $\lambda > 0$.

(b) (Converse part) $\limsup_{n\to\infty}(1/n)\log K(n,\lambda) \leq R^*(V,W)$ *for all* $\lambda < 1$.

Thus, $\lim_{n\to\infty}(1/n)\log K(n,\lambda) = R^*(V,W)$ *for all* $\lambda \in (0,1)$.

Define rate R of generating common randomness to be achievable if there exists a sequence of (n, K_n, λ_n) protocols such that $\lim \lambda_n = 0$ and $\liminf (1/n)\log K_n \geq R$. Then, Part (a) of Theorem 2.1 is obviously equivalent to the statement that any rate not exceeding R^* is achievable. A "weak" converse to this result would only assert that rates above R^* are not achievable, i.e., $\lim_{\lambda\downarrow 0} \limsup_{n\to\infty}(1/n)\log K(n,\lambda) \leq R^*$. However, Part (b) of Theorem 2.1 says much more than this — in the usual terminology, it is a "strong" converse.

Theorem 2.1 justifies the interpretation of $R^*(V,W)$ as the *common randomness capacity* of the given pair of channels, in the absence of external sources. We will prove the direct part in Section 3, and the converse part in Section 4.

2.1 Binary symmetric channels

Suppose V and W are both binary symmetric channels, with crossover probabilities p and q respectively. Then, the common randomness capacity can be shown to equal $\min\{h(p) + h(q), 2 - h(p) - h(q)\}$, where $h(\cdot)$ is the binary entropy function. Note that this is 0 if and only if $h(p) = h(q) = 0$ or $h(p) = h(q) = 1$. In the first case, the two channels do not provide any randomness (zero entropy), although they allow for perfect agreement between the two agents (high capacity). In the second case, the situation is reversed; a transmission by either agent provides a totally random bit to the other (high entropy), but the randomness generated this way cannot be reliably communicated back to the sender (zero capacity).

On the other hand, the capacity attains its maximum value of 1 whenever the entropies and capacities balance each other, i.e., when $h(p) + h(q) = (1 - h(p)) + (1 - h(q))$. It is somewhat surprising that it is possible to generate common randomness at a rate of 1 bit per step in all these cases.

3 Proof of the direct part

To prove the direct part, it suffices to show that, for any $R < R^*$, there exist (t^2, K_t, λ_t) protocols for all $t \geq 1$, with $\lim_{t\to\infty} \lambda_t = 0$ and $\liminf_{t\to\infty}(1/t^2)\log K_t \geq R$. For, given any n satisfying $t^2 < n < (t+1)^2$, Alice and Bob could execute the (t^2, K_t, λ_t) protocol and fill the remaining $n - t^2$ steps arbitrarily, without affecting the rate achieved. The following lemma will be crucial in the proof.

Lemma 3.1 *Let $(\mathcal{Z}, \hat{\mathcal{Z}}, U)$ be any DMC, $d = |\mathcal{Z}||\hat{\mathcal{Z}}|$, and $P \in \mathcal{P}_t(\mathcal{Z})$ for some $t \geq 1$.*

(1) If $R' \geq 0$ and $J \leq \exp(tR')$, then there exist sequences $\mathbf{c}_1, \ldots, \mathbf{c}_J$ in $\mathcal{Z}^t$, all of type P, and a partition of $\hat{\mathcal{Z}}^t$ into subsets $\mathcal{C}_1, \ldots, \mathcal{C}_J$, such that $1 - U^t(\mathcal{C}_j|\mathbf{c}_j) \leq \exp\{-tE'(R', P, U) + o(t)\}$ for all $j \in [J]$. Here, $E'(R', P, U)$ is a continuous function of (R', P) which is positive if $R' < I(P;U)$ and zero otherwise.

(2) If $R'' \geq 0$ and $L \leq \exp(tR'')$, then for any $\mathbf{c} \in \mathcal{Z}^t$ of type P, and any $\mathcal{C} \subseteq \hat{\mathcal{Z}}^t$, there exists a partition of $\mathcal{C}$ into subsets $\mathcal{C}_e, \mathcal{C}_1, \ldots, \mathcal{C}_L$, such that $U^t(\mathcal{C}_l|\mathbf{c}) = (1/L)[U^t(\mathcal{C}|\mathbf{c}) - U^t(\mathcal{C}_e|\mathbf{c})]$ for all $l \in [L]$, and $U^t(\mathcal{C}_e|\mathbf{c}) \leq \exp\{-tE''(R'', P, U) + o(t)\}$. Here, $E''(R'', P, U)$ is a continuous function of (R'', P) which is positive if $R'' < H(U|P)$ and zero otherwise.

Proof: Part (1) is a standard result in channel coding. See Theorem 5.2 on p. 165 of [6] for a proof. Part (2) can be proved by simple type-based arguments. For details, see [10]. □

Let $P^* \in \mathcal{P}(\mathcal{X})$ and $Q^* \in \mathcal{P}(\mathcal{Y})$ be the distributions achieving the maximum in (2). Let

$$\begin{aligned} R_A &= \min[H(W|Q^*), I(P^*;V)] - \epsilon, \\ R_B &= \min[H(V|P^*), I(Q^*;W)] - \epsilon. \end{aligned}$$

Let $M_t = \lfloor \exp(tR_A) \rfloor - 1$ and $N_t = \lfloor \exp(tR_B) \rfloor - 1$. We will define a $(t^2, (M_t N_t)^{t-1}, \lambda_t)$ protocol for each $t \geq 1$, with $\lim_{t\to\infty} \lambda_t = 0$. This sequence of protocols will then achieve the rate $R_A + R_B = R^* - 2\epsilon$.

Choose types $P_t \in \mathcal{P}_t(\mathcal{X})$ and $Q_t \in \mathcal{P}_t(\mathcal{Y})$, $t = 1, 2, \ldots$, such that $P_t \to P^*$ and $Q_t \to Q^*$ as $t \to \infty$. Then, by Lemma 3.1:

(a) there exist sequences $\mathbf{a}_i \in \mathcal{X}^t$ of type P_t, and sets $\mathcal{A}_i$ partitioning $\hat{\mathcal{X}}^t$ and satisfying $1 - V^t(\mathcal{A}_i|\mathbf{a}_i) \leq \alpha_t = \exp\{-tE'(R_A, P_t, V) + o(t)\}$, for each $i \in \{e\} \cup [M_t]$. Further, each $\mathcal{A}_i$ can be partitioned into subsets $\mathcal{A}_i(j')$, $j' \in \{e\} \cup [N_t]$, such that $V^t(\mathcal{A}_i(j')|\mathbf{a}_i)$ is the same for all $j' \in [N_t]$, and $V^t(\mathcal{A}_i(e)|\mathbf{a}_i) \leq \rho_t = \exp\{-tE''(R_B, P_t, V) + o(t)\}$.

(b) there exist sequences $\mathbf{b}_j \in \mathcal{Y}^t$ of type Q_t, and sets $\mathcal{B}_j$ partitioning $\hat{\mathcal{Y}}^t$ and satisfying $1 - W^t(\mathcal{B}_j|\mathbf{b}_j) \leq \beta_t = \exp\{-tE'(R_B, Q_t, W) + o(t)\}$, for each $j \in \{e\} \cup [N_t]$. Further, each $\mathcal{B}_j$ can be partitioned into subsets $\mathcal{B}_j(i')$, $i' \in \{e\} \cup [M_t]$, such that $W^t(\mathcal{B}_j(i')|\mathbf{b}_j)$ is the same for all $i' \in [M_t]$, and $W^t(\mathcal{B}_j(e)|\mathbf{b}_j) \leq \sigma_t = \exp\{-tE''(R_A, Q_t, W) + o(t)\}$.

Also, pick arbitrary sequences $\mathbf{a} \in \mathcal{X}^t$ and $\mathbf{b} \in \mathcal{Y}^t$ of types P_t and Q_t respectively. Then, by Part (2) of Lemma 3.1, (a) $\hat{X}^t$ can be partitioned into subsets $\mathcal{A}(j')$, $j' \in \{e\} \cup [N_t]$, such that $V^t(\mathcal{A}(j')|\mathbf{a})$ is the same for all $j' \in [N_t]$, and $V^t(\mathcal{A}(e)|\mathbf{a}) \leq \rho_t$; (b) $\hat{Y}^t$ can be partitioned into subsets $\mathcal{B}(i')$, $i' \in \{e\} \cup [M_t]$, such that $W^t(\mathcal{B}(i')|\mathbf{b})$ is the same for all $i' \in [M_t]$, and $W^t(\mathcal{B}(e)|\mathbf{b}) \leq \sigma_t$.

3.1 Outline of the protocol

The t^2-step protocol proceeds in t *rounds*, indexed $0, 1, \ldots, t-1$, in each of which Alice and Bob send each other sequences of length t. We will describe these t rounds recursively.

In round 0, Alice and Bob transmit the sequences $\mathbf{a}$ and $\mathbf{b}$ respectively. Alice defines S_1 to be the $i \in \{e\} \cup [M_t]$ such that the sequence she received from Bob falls in $\mathcal{B}(i)$. Similarly, Bob defines T_1 to be the $j \in \{e\} \cup [N_t]$ such that the sequence he received from Alice falls in $\mathcal{A}(j)$. This completes round 0.

Now let $1 \leq k < t$. Assume that Alice and Bob have computed $S_k \in \{e\} \cup [M_t]$ and $T_k \in \{e\} \cup [N_t]$, respectively, based on the sequences they received in round $k-1$. Then, in round k, Alice transmits the codeword $\mathbf{a}_{S_k}$ and Bob transmits the codeword $\mathbf{b}_{T_k}$.

Based on the sequences they receive from each other, Alice and Bob try to guess the index of the codeword sent by the other, and also decide the index of the codeword to transmit in the *next* round. This is done as follows: Alice finds the $(i,j) \in (\{e\} \cup [M_t]) \times (\{e\} \cup [N_t])$ such that the sequence she received falls in $\mathcal{B}_j(i)$. She then estimates T_k as $\hat{T}_k = j$, and takes $S_{k+1} = i$. Similarly, Bob finds the $(i,j) \in (\{e\} \cup [M_t]) \times (\{e\} \cup [N_t])$ such that the sequence he received falls in $\mathcal{A}_i(j)$. He then estimates S_k as $\hat{S}_k = i$, and takes $T_{k+1} = j$. This completes round k.

Let

$$\mathbf{S} = \Big((S_1, \hat{T}_1), (S_2, \hat{T}_2), \ldots, (S_{t-1}, \hat{T}_{t-1})\Big),$$

$$\mathbf{T} = \Big((\hat{S}_1, T_1), (\hat{S}_2, T_2), \ldots, (\hat{S}_{t-1}, T_{t-1})\Big).$$

Both $\mathbf{S}$ and $\mathbf{T}$ can take on $((M_t+1)(N_t+1))^{t-1}$ different values. Of these, there are $(M_t N_t)^{t-1}$ in which none of the $2(t-1)$ components is e. Let $\mathcal{R}$ be an arbitrary function that maps these $(M_t N_t)^{t-1}$ possibilities onto $[(M_t N_t)^{t-1}]$, and maps all the remaining possibilities to e. Then, after round $t-1$, Alice and Bob take their random outputs to be $S = \mathcal{R}(\mathbf{S})$ and $T = \mathcal{R}(\mathbf{T})$, respectively. Thus, both S and T take values in $\{e\} \cup [(M_t N_t)^{t-1}]$. It can be proved that the "error probability" λ_t associated with the above protocol goes to 0 as $t \to \infty$.

Claim 3.1 *For each $k \in \{1, 2, \ldots, t-1\}$, choose any $(i_k, j_k) \in [M_t] \times [N_t]$. Then,*

$$\frac{1}{(M_t N_t)^{t-1}} \;\geq\; Pr\Big\{\mathbf{S} = \mathbf{T} = ((i_k, j_k))_{k=1}^{t-1}\Big\} \;\geq\; \frac{1-\lambda_t}{(M_t N_t)^{t-1}}; \tag{3}$$

here, $\lambda_t = t\,(\alpha_t + \rho_t + \beta_t + \sigma_t) \to 0$ as $t \to \infty$.

Proof: See [10]. □

4 Proof of the converse part

Let (f, g, S, T) be any (n, K, λ) protocol, with $\lambda < 1$. We will prove that $K \leq 2^{nR^* + o(n)}$, by exhibiting a $P \in \mathcal{P}(\mathcal{X})$ and a $Q \in \mathcal{P}(\mathcal{Y})$ such that $(1/n) \log K$ is bounded above by

$$\min\Big\{H(PV), H(QW), H(V|P) + H(W|Q), I(P;V) + I(Q;W)\Big\} + o(1). \tag{4}$$

The minimum in (4) equals $\min[H(W|Q), I(P;V)] + \min[H(V|P), I(Q;W)]$. The key idea in the proof is Lemma 4.1, which helps identify a suitable high-probability subset of "jointly typical" $(\hat{x}^n, \hat{y}^n)$ sequences.

First some notation: for any $(\hat{x}^n, \hat{y}^n)$, let $N(\hat{x}|\hat{x}^n) = |\{k : \hat{x}_k = \hat{x}\}|$, $N(\hat{y}|\hat{y}^n) = |\{k : \hat{y}_k = \hat{y}\}|$, $N(x|\hat{y}^n) = |\{k : f_k(\hat{y}^{k-1}) = x\}|$, $N(y|\hat{x}^n) = |\{k : g_k(\hat{x}^{k-1}) = y\}|$, $N(x, \hat{x}|\hat{x}^n, \hat{y}^n) = |\{k : f_k(\hat{y}^{k-1}) = x, \hat{x}_k = \hat{x}\}|$, and $N(y, \hat{y}|\hat{x}^n, \hat{y}^n) = |\{k : g_k(\hat{x}^{k-1}) = y, \hat{y}_k = \hat{y}\}|$. Denote $Pr\{\hat{X}^n = \hat{x}^n, \hat{Y}^n = \hat{y}^n\}$ under (f, g) by $U_{f,g}(\hat{x}^n, \hat{y}^n)$. Also, write $H^V(x)$ and $H^W(y)$ for $H(V(\cdot|x))$ and $H(W(\cdot|y))$.

Lemma 4.1 *Let $\mathcal{E}$ be the set of all $(\hat{x}^n, \hat{y}^n) \in \hat{\mathcal{X}}^n \times \hat{\mathcal{Y}}^n$ such that:*

(a) $|N(\hat{x}|\hat{x}^n) - \sum_x N(x|\hat{y}^n)V(\hat{x}|x)| \le \theta\sqrt{n}$ *for all* $\hat{x} \in \hat{\mathcal{X}}$.

(b) $\left|N(\hat{y}|\hat{y}^n) - \sum_y N(y|\hat{x}^n)W(\hat{y}|y)\right| \le \theta\sqrt{n}$ *for all* $\hat{y} \in \hat{\mathcal{Y}}$.

(c) $\left|\log U_{f,g}(\hat{x}^n, \hat{y}^n) + \sum_x N(x|\hat{y}^n)H^V(x) + \sum_y N(y|\hat{x}^n)H^W(y)\right| \le \theta\sqrt{n}$.

Then, $1 - U_{f,g}(\mathcal{E}) \le \gamma/\theta^2$, where γ is a constant determined by V and W.

Proof: Essentially by Chebyshev's inequality. See [10] for details. □

Assume from now on that θ is so large that $U_{f,g}(\mathcal{E}) \ge (1+\lambda)/2$. For $l \in [K]$, let $\mathcal{E}_l = \{\hat{x}^n : T(\hat{x}^n) = l\} \times \{\hat{y}^n : S(\hat{y}^n) = l\}$. Then, $U_{f,g}(\mathcal{E} \cap \bigcup_{l\in[K]} \mathcal{E}_l) \ge (1-\lambda)/2$.

If σ' is an n-type on $\mathcal{X} \times \hat{\mathcal{X}}$ and τ' is an n-type on $\mathcal{Y} \times \hat{\mathcal{Y}}$, let $\mathcal{E}_{\sigma',\tau'}$ be the set of $(\hat{x}^n, \hat{y}^n) \in \mathcal{E}$ such that $N(x, \hat{x}|\hat{x}^n, \hat{y}^n) = n\sigma'(x, \hat{x})$ and $N(y, \hat{y}|\hat{x}^n, \hat{y}^n) = n\tau'(y, \hat{y})$ for all $x, \hat{x}, y, \hat{y}$. Since the number of such (σ', τ') pairs is bounded above by n^c for a suitably large c, and $U_{f,g}(\mathcal{E} \cap \bigcup_{l\in[K]} \mathcal{E}_l) \ge (1-\lambda)/2$, there must exist a pair (σ, τ) such that $U_{f,g}(\mathcal{E}_{\sigma,\tau} \cap \bigcup_{l\in[K]} \mathcal{E}_l) \ge (1-\lambda)/2n^c$. Let $P_\sigma(x) = \sum_{\hat{x}} \sigma(x, \hat{x})$, $Q_\tau(y) = \sum_{\hat{y}} \tau(y, \hat{y})$, $\hat{P}_\sigma(\hat{x}) = \sum_x \sigma(x, \hat{x})$, and $\hat{Q}_\tau(\hat{y}) = \sum_y \tau(y, \hat{y})$ be the marginals of σ and τ. The idea is to prove (4) with P_σ and Q_τ in place of P and Q. Note that, by condition (a) in the definition of $\mathcal{E}$, $|\hat{P}_\sigma(\hat{x}) - P_\sigma V(\hat{x})| \le \theta n^{-1/2} = o(1)$ for all $\hat{x}$, so that $|H(\hat{P}_\sigma) - H(P_\sigma V)| = o(1)$ by the continuity of the entropy function. Similarly, $|H(\hat{Q}_\tau) - H(Q_\tau W)| = o(1)$.

It can easily be shown that

$$|\mathcal{E}_{\sigma,\tau}| \le \exp\{n[H(V|P_\sigma) + H(W|Q_\tau)] + o(n)\}. \tag{5}$$

Also,

$$|\{\hat{x}^n : \exists\, \hat{y}^n \text{ s.t. } (\hat{x}^n, \hat{y}^n) \in \mathcal{E}_{\sigma,\tau}\}| \le \exp\{nH(P_\sigma V) + o(n)\}; \tag{6}$$

$$|\{\hat{y}^n : \exists\, \hat{x}^n \text{ s.t. } (\hat{x}^n, \hat{y}^n) \in \mathcal{E}_{\sigma,\tau}\}| \le \exp\{nH(Q_\tau W) + o(n)\}. \tag{7}$$

Finally, if $(\hat{x}^n, \hat{y}^n) \in \mathcal{E}_{\sigma,\tau}$, then it can be shown that

$$n^{-1} \log \left(\frac{U_{f,g}(\hat{x}^n, \hat{y}^n)}{\hat{P}_\sigma^n(\hat{x}^n)\hat{Q}_\tau^n(\hat{y}^n)} \right) = I(P_\sigma; V) + I(Q_\tau; W) + o(1). \tag{8}$$

Now, many of the decision regions $\mathcal{E}_l$ must intersect significantly with $\mathcal{E}_{\sigma,\tau}$. More precisely, if

$$\mathcal{L} = \{l \in [K] : U_{f,g}(\mathcal{E}_{\sigma,\tau} \cap \mathcal{E}_l) \ge (1-\lambda)/4Kn^c\},$$

then it can be proved that $|\mathcal{L}| \ge K(1-\lambda)/8n^c$. But if $l \in \mathcal{L}$ then $U_{f,g}(\mathcal{E}_{\sigma,\tau} \cap \mathcal{E}_l) > 0$, so that $\mathcal{E}_{\sigma,\tau} \cap \mathcal{E}_l$ must be non-empty. Therefore,

(a) $|\mathcal{L}| \leq |\mathcal{E}_{\sigma,\tau}| \leq \exp\{n[H(V|P_\sigma) + H(W|Q_\tau)] + o(n)\}$.

(b) $|\mathcal{L}| \leq |\{\hat{x}^n : \exists\, \hat{y}^n \text{ s.t. } (\hat{x}^n, \hat{y}^n) \in \mathcal{E}_{\sigma,\tau}\}| \leq \exp\{nH(P_\sigma V) + o(n)\}$.

(c) $|\mathcal{L}| \leq |\{\hat{y}^n : \exists\, \hat{x}^n \text{ s.t. } (\hat{x}^n, \hat{y}^n) \in \mathcal{E}_{\sigma,\tau}\}| \leq \exp\{nH(Q_\tau W) + o(n)\}$.

The right inequalities in (a), (b), and (c) are by (5), (6), and (7) respectively. Together with the lower bound on $|\mathcal{L}|$, these inequalities yield three of the desired terms in the minimum occuring in (4). The fourth term can be proved using (8), together with an application of the Cauchy-Schwarz inequality. Details can be found in [10]. □

References

[1] R. Ahlswede. Elimination of correlation in random codes for arbitrarily varying channels. *Z. Wahrsch. Verw. Gebiete*, 33:159–175, 1978.

[2] R. Ahlswede and I. Csiszàr. Common randomness in information theory and cryptography - part I: Secret sharing. *IEEE Transactions on Information Theory*, Vol. 39(No. 4), July 1993.

[3] R. Ahlswede and G. Dueck. Identification in the presence of feedback - a discovery of new capacity formulas. *IEEE Transactions on Information Theory*, Vol. 35(No. 1), January 1989.

[4] R. Ahlswede and G. Dueck. Identification via channels. *IEEE Transactions on Information Theory*, Vol. 35(No. 1), January 1989.

[5] R. Ahlswede and B. Verboven. On identification via multiway channels with feedback. *IEEE Transactions on Information Theory*, Vol. 37(No. 5), September 1991.

[6] I. Csiszar and J. Korner. *Information Theory: Coding Theorems for Discrete Memoryless Systems.* Academic Press, 1981.

[7] I. Csiszar and P. Narayan. The capacity of the arbitrarily varying channel revisited: Positivity constraints. *IEEE Transactions on Information Theory*, 34:181–193, March 1988.

[8] L. Lovász. Communication complexity: A survey. In B.H. Korte et al., editors, *Paths, Flows and VLSI layout.* Springer-Verlag, 1990.

[9] A. Orlitsky and A. El Gamal. Communication complexity. In Y. Abu-Mostafa, editor, *Complexity in Information Theory.* Springer-Verlag, 1988.

[10] S. Venkatesan and V. Anantharam. The common randomness capacity of a pair of independent discrete memoryless channels. Technical Report UCB/ERL M95/85, Electronics Research Laboratory, Univ. of California, Berkeley, September 1995.

Transmission Energy Allocation for Robust Multi-Stage Vector Quantization*

Shrinivas Gadkari and Kenneth Rose
Center for Information Processing Research
Department of Electrical and Computer Engineering
University of California, Santa Barbara, CA 93106

Abstract

We attack the problem of robust vector quantization over time varying noisy channels. For practical considerations of computational and storage complexity, we restrict the discussion to multi-stage vector quantizers. We show that robustness to channel errors can be achieved by allocating transmission energy to the different stages of the quantizer according to their importance. The contributions of this paper include: algorithms for transmission energy allocation to different stages, properties of the optimal solution and design of decoders that adapt to the time-varying conditions of the channel. Simulation results compare the performance of energy allocation with that of index assignment and demonstrate substantial gains over widely varying channel conditions, at the cost of moderate to no increase in complexity. Extensions to the case of time-varying channels with known statistics is discussed.

1 Introduction

Vector quantization has become a popular and effective technique for signal compression and is used as the central building block in an increasing number of speech and image coding algorithms [8]. With the advent of wireless communication systems, there has been a growing realization of the vital need to effectively protect the compressed signal against channel errors. Several source-channel coding methods have been developed in recent years (e.g., [3, 4]). Most of the reported techniques optimize the vector quantizer (VQ) while assuming a fixed modulation scheme. However, as will be demonstrated here, by optimizing the modulation parameters for the given source and VQ structure we can achieve increased gains over a large range of channel conditions.

VQ design techniques for noisy channels broadly fall into two classes. The first category of techniques aim at imparting robustness to the VQ without sacrificing the performance under noiseless channel conditions. The primary reason for this being

*This work was supported in part by National Science Foundation under grant no. NCR-9314335, University of California MICRO program, Cisco Systems, DSP Group, Inc., DSP Software Engineering, Echo Speech Corp., General Electric, Hughes Electronics, Intel, Moseley Associates, Qualcomm, Inc., Rockwell International Corp., Speech Technology Labs, and Texas Instruments, Inc.

the lack of precise knowledge about the channel characteristics (which might be time varying). Here we design the VQ optimized for a noiseless channel and employ techniques like index assignment to impart robustness to the VQ against channel errors, e.g. [4, 3]. The second class of design techniques are channel specific. Here we have some information about the statistics of the noise in the channel which is incorporated in the VQ design, e.g. [3, 5]. The techniques described in this paper mostly fall under the former category. We however present some preliminary ideas about VQ design for time-varying channels with known statistics.

We focus our attention on a multi-stage VQ (MSVQ) which is widely used in applications (notably in speech coding) due to its reduced storage and encoding complexity. In this paper we present a new approach to combined source-channel coding whose objective is to provide adequate channel protection to the different stages of the MSVQ according to their relative importance, while maintaining close to optimal performance when noiseless channel conditions prevail. The main idea is to assign varying levels of transmission energy to various stages according to their importance. This technique has its roots in the 1958 paper of Bedrosian [1], and the later work of Sundberg (see recent description in [2]), where bitwise allocation of transmission energy was suggested for pulse code modulation. We first develop the transmission energy allocation (TEA) approach to MSVQ design for noisy channels, and then provide experimental results to support its superior performance. We further extend the use of TEA with an adaptive decoder (that has access to information on rapidly varying channel conditions). The results show much promise for this approach where substantial gains are available at the cost of moderate to no increase in complexity. Finally we consider extensions to the case of a time-varying channel with known statistics.

2 Multi-Stage VQ - Basic Structure

An N-stage MSVQ consists of N codebooks, $C_1, C_2, .., C_N$, where codebook C_i is a set of 2^{k_i} codevectors addressable with k_i bits. An M-L search encoding algorithm proceeds by comparing the source vector x with all the code vectors in C_1 and choosing the L best vectors that approximate x in the least square sense. These are the L survivors at stage 1 and will be denoted by $\{y^{(l)}{}_1\}$. In the next step, we consider all possible combinations of these L survivors with the codevectors in C_2 and choose the L combinations $\{y^{(l)}{}_{12}\}$, that best approximate x, these are the survivors at stage 2. We repeat this procedure of selecting selecting L survivors at each stage. For the N th stage however, we select only one combination that best approximates x. The resulting quantized version of x is given by

$$y = y_1 + y_2 + \cdots + y_N, \qquad \text{where } y_i \in C_i. \tag{1}$$

The indices or addresses of the selected codevectors denoted by $\{I_1, I_2, ..., I_N\}$ are transmitted over a Gaussian channel using binary modulation[1] Let $\{J_1, J_2, ..., J_N\}$ be the

[1]In this paper we discuss TEA in the context of binary modulation. The idea is however applicable to any coded modulation scheme. See [9] for a discussion of TEA applied to coded modulation.

i	1	2	3	4
D_i	0.194	0.035	0.009	0.002

Table 1: The partial distortion for various stages of a 4-stage MSVQ. For all the stages, $\epsilon_i = 0.01$. The VQ dimension was 4, and it was designed for a first order Gauss-Markov source with unit variance and correlation = 0.5.

indices received by the decoder. The decoder produces a reproduction y' given by

$$y' = v_1(J_1) + v_2(J_2) + \cdots + v_N(J_N). \tag{2}$$

For now we restrict the discussion to a simple (non-adaptive) decoder where $v_i(J_i)$ is simply the J_ith entry in codebook C_i. We will later consider improved, adaptive decoders.

We define the distortion due to channel errors, D_c, as

$$D_c = E||y - y'||^2. \tag{3}$$

We simplify D_c by neglecting the cross terms to obtain,

$$D_c = \sum_{i=1}^{N} D_i,$$

where, $D_i = E||y_i - v_i||^2$. This approximation is reasonable, especially whenever channel errors in the various stages are independent and approximately zero-mean as is often the case. Let the ith stage be transmitted on a channel with probability of bit error ϵ_i, then the stage distortion is

$$D_i = \sum_I P_i(I) \sum_J ||y_i(I) - y_i(J)||^2 \epsilon_i^{d_H(I,J)} (1 - \epsilon_i)^{k_i - d_H(I,J)}, \tag{4}$$

where, $y_i(I)$ is the code vector indexed by I in code book C_i, $P_i(I)$ is the probability that index I is transmitted and $d_H(I, J)$ is the Hamming distance between the binary representation of indices I and J.

The effect of channel errors on MSVQ stages is highly nonuniform. This fact is demonstrated in Table 1 which lists the partial distortions D_i for various stages of a 4-stage VQ, transmitted under the same channel conditions. Clearly, an unequal error protection scheme should be able to exploit this feature of MSVQ by reducing the transition probability of more sensitive stages at the cost of increasing the transition probability of the less sensitive stages. To achieve this objective we propose the use of a simple yet very effective scheme of *transmission energy allocation* (TEA).

3 Transmission Energy Allocation

Our main purpose is to achieve robust performance over a range of channel conditions. Table 1 clearly suggests that for a given level of channel noise, earlier MSVQ stages need more protection than later stages. We choose a level of channel noise that represents the

average channel conditions and provide a varying level of protection to MSVQ stages so as to obtain an optimal performance at the given level of channel noise.

We assume a transmission scheme where each of the N stages is transmitted independently. In other words, any form of coded modulation would be done by grouping bits of the same stages from consecutive codewords. Although this paper will target uncoded systems, the formulation is kept general to allow its application to systems that use some form of explicit coded modulation, and in fact includes the uncoded system as a special case.

We are given a quota of transmission energy E_{tot} to be distributed to the N stages. Let E_i denote the energy allocated to the i th stage. The probability of bit error for the i th stage can be written as $\epsilon_i = erfc\left(\frac{E_i}{k_i \sigma_n^2}\right)$, where σ_n^2 is the variance of the Gaussian channel noise. We wish to minimize D_c over all choices of $\{E_i\}$ subject to the total energy constraint: [2]

$$\sum_{i=1}^{N} E_i = E_{tot}.$$

From the corresponding Lagrangian we get the necessary condition for optimality:

$$\frac{\partial D_c}{\partial E_i} = \frac{\partial D_i}{\partial E_i} = -\lambda, \qquad \text{if } E_i > 0, \tag{5}$$

where, λ is the Lagrange multiplier whose value is chosen to satisfy the total energy constraint. Although one can easily derive various descent algorithms, we chose to describe here an algorithm that is analogous to known bit allocation algorithms (e.g., [8]). But before describing the algorithm, let us gain further insights into the solution to the energy allocation problem.

3.1 Properties of Optimal TEA

An important aspect of the optimal TEA is the existence of a "cutoff" stage energy level (depending on the representative level of channel noise) below which it is better not to transmit the stage at all. In this section we discuss this case along with its implications on the energy allocation solution.

First let us consider the decoding strategy when some stage is not transmitted. Let the decoder (that knows the transmission strategy) use a reproduction μ_i, when the i th stage is not transmitted. The optimal value of μ_i minimizes $E||y_i - \mu_i||^2$ and is given by $\mu_i = E[y_i]$. The resulting distortion is called the cutoff distortion D_i^* and can be obtained with zero energy allocated to the ith stage. Hence we have

$$D_i^* = E||y_i - \mu_i||^2. \tag{6}$$

A typical plot of $D_i(E_i)$ for a non-adaptive decoder is shown in Figure 1 (please ignore the adaptive decoder curve for now). We can see that $D_i^* < \lim_{E_i \to 0} D_i(E_i)$. Let E_i^* be the

[2]The actual quantity of interest to us is $E||x-y'||^2$. We approximate this quantity as $E||x-y||^2+D_c$. This approximation becomes exact when y is the centroid of the corresponding quantization cell in the input space (optimal VQ). Hence we choose to minimize D_c.

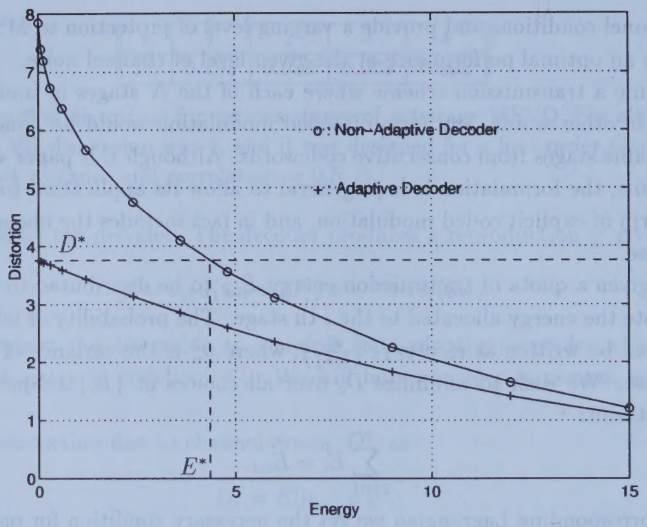


Figure 1: A plot of typical behavior of $D_i(E_i)$ for an adaptive and non-adaptive decoder.

cutoff energy E_i^*, that is, the value of energy such that $D_i^* = D_i(E_i^*)$. The significance of E_i^* is that stage transmission is wasteful unless we allocate energy $E_i > E_i^*$ to it. In other words, if some stage is transmitted using energy less than E_i^*, D_c could be reduced by not transmitting this stage at all. Moreover, this "leftover" energy can be re-allocated to other stages and reduce D_c further.

We say that an energy allocation solution is *monotonic* if the stage transmission energy sequence is non-increasing; and "well behaved" if $E_i = 0$ implies $E_{i+1} = 0, E_{i+2} = 0, ..., E_N = 0$. Of course, monotonic solutions are well behaved. We will restrict the search to well behaved solutions.

Remark: If zero energy is allocated to some stages, we no longer have the same noiseless channel performance. This is due to the fact that ${\sigma_n}^2$ - the representative level of channel noise, is too high for transmission of all stages with the available transmission energy, E_{tot}.

3.2 The Energy Allocation Algorithm

In this section we describe a greedy energy allocation algorithm. We start by choosing a small quantum of energy ΔE, such that $E_{tot} = T\Delta E$, for some integer T. The energy allocated to each stage will be constrained to be an integer multiple of ΔE. Let t_i denote the quanta of energy allocated to the ith stage, i.e. $E_i = t_i \Delta E$. Our problem reduces to allocating the available T quanta of energy to the N stages, while satisfying the constraint $\sum_i t_i = T$. To simplify the description of the algorithm, let us introduce the quantities $\{t_i^*\}$, which denote the minimal quanta of energy to be allocated to the ith stage in order to enable its transmission: $t_i^* = \left\lceil \frac{E_i^*}{\Delta E} \right\rceil$.

(1) Set $i = 1$, and allocate quanta t_1^* to the first stage.
(2) If the amount of energy available to be allocated exceed t_{i+1}^*, execute step (2a), else execute step (2b).
(2a) Compare $\frac{\partial D_j}{\partial E_j}$ for $j = 1, 2, ..., i$ and $\frac{\partial D_{i+1}}{\partial E_{i+1}}$ evaluated at E_{i+1}^*. If the stage $i+1$ has the larger value of $|\frac{\partial D_{i+1}}{\partial E_{i+1}}|$, allocate t_{i+1}^* quanta of energy to the $i+1$ th stage. If one of other stages has the larger value of $|\frac{\partial D_j}{\partial E_j}|$, allocate a single quantum of energy to that stage.
(2b) Compare $\frac{\partial D_j}{\partial E_j}$ for $j = 1, 2, ..., i$, and allocate energy a quantum of energy to the stage with the largest value of $|\frac{\partial D_j}{\partial E_j}|$.
(3) Update the quanta of energy available to be allocated. If energy was allocated to the $i+1$ th stage in step (2), increment i. Go to step (2).
(4) Stop when all the quanta of energy have been allocated.

4 Adaptive Decoder

The decoder that we employed in the last section simply produced an output by addressing the entry in the codebook addressed by the received index. However if the receiver has access to the current level of channel noise (which is possible in many applications where the detection scheme produces some estimate of the channel SNR) an improved performance can be achieved by employing a more intelligent decoder, the adaptive decoder.

Let S denote the present state of the channel. The decoder receives indices $\{J_1, J_2, .., J_N\}$ whose corresponding reproduction vectors are $\{v_1(J_1), v_2(J_2), \ldots, v_N(J_N)\}$. We now find the optimal v_i by minimizing $E[\|v_i - y_i\|^2 | J_i, S]$:

$$v_i(J_i) = \frac{\sum_I p(J_i|I,S)P_i(I)y_i(I)}{\sum_I p(J_i|I,S)P_i(I)}. \tag{7}$$

Since the state information S is available to the decoder, it can evaluate $v_1, v_2, \ldots, v_M$ using (7). Note that the complexity involved in the computation of v_i is proportional to 2^{k_i}, the size of the codebook C_i. Hence, the complexity involved in the reproduction of v_i at the decoder is of the same order as the computation involved in the ith codebook search at the encoder. One can therefore safely assume that the complexity of the adaptive decoder is manageable whenever the encoding complexity is.

For the case of an adaptive decoder we have, $D_i = E||y_i - v_i||^2$. The typical behavior of $D_i(E_i)$, for an adaptive decoder is shown in Figure 1. For this case, we have $D_i^* = \lim_{E_i \to 0} D_i(E_i)$. Thus the adaptive decoder has cutoff energy $E_i^* = 0$, and the energy allocation algorithm simplifies to :

(1) Set $I = 1$, and allocate energy ΔE to stage 1.

(2) Compare $\frac{\partial D_c}{\partial E_i}$ for $i = 1, 2, ..., I$ and $\frac{\partial D_c}{\partial E_{I+1}}$ evaluated at $E_{I+1} = 0$, where, estimates of the adaptive decoder are based on the representative channel condition. Allocate energy ΔE to the stage with the largest value of $|\frac{\partial D_c}{\partial E_i}|$. Update the quota of energy available to be allocated.

(3) If energy was allocated to the $I+1$ th stage in step (2), increment I. Go to step (2).

(4) Stop when all the energy has been allocated.

The advantages of the adaptive decoder over the nonadaptive decoder are obviously emphasized at very noisy channel conditions, where it does not suffer from cutoff problems. To gain further intuition on the superiority of the adaptive decoder, consider the case when the optimal solution at the representative level of channel noise requires in the nonadaptive case not to transmit some stages of the MSVQ. In the adaptive decoder case, however, instead of not transmitting these stages, we transmit them with a small amount of energy. When the level of channel noise is high, the distortion incurred is close to D_i^* - the same distortion had we not transmitted those stages. However when the level of channel noise at the decoder drops, the adaptive decoder can gain significantly from this information. This is not possible with the non-adaptive decoder since, for this decoder $\lim_{E_i \to 0} D_i(E_i) > D_i^*$, and transmitting some stage with a small amount of energy causes more loss than not transmitting that stage when the level of channel noise is high.

5 Results

A first order Gauss-Markov process with correlation coefficient 0.5 was quantized using a 4-dimensional MSVQ. A 4-stage VQ with codebooks of size 64 for all the stages was designed. The performance of the system for channel SNR ranging from 0 dB to 10 dB was studied. The schemes compared were : (1) IA scheme : Here we design the 4-stage MSVQ and apply index assignment on the codebooks of all the stages of the VQ. (2) TEA scheme : Here we use the same codebooks (with index assignment) as used by the IA scheme. Transmission energy allocation is done for the case of both non-adaptive and adaptive decoder. The TEA was optimized for channel SNR of 9 dB for both type of receivers. The performance of these systems over a range of channel conditions was evaluated by simulation. The resulting overall SNR - $\frac{E||x||^2}{E||x-y'||^2}$, as a function of the Gaussian channel SNR is plotted in Figure 2.

We make the following observations:

(1) Compared to IA, TEA achieves performance gains of over 4 dB. Note that although TEA is optimized for a channel SNR of 9 dB, this system outperforms IA at all the levels of channel SNR.

(2) The advantages of the adaptive decoder over the non-adaptive decoder start becoming evident under conditions of high channel noise. The reasons for this were briefly discussed in section 4.

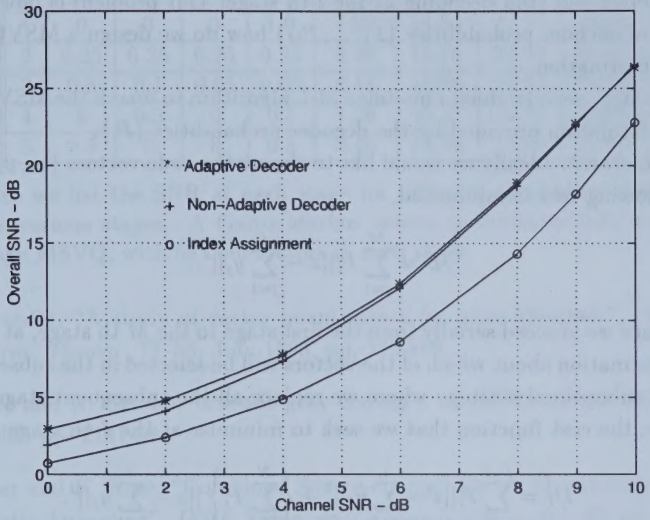


Figure 2: Here we compare the performance of TEA with index assignment (IA). The TEA results are given for the case of adaptive and non-adaptive decoder.

6 Extensions and Discussion

Here we discuss some preliminary ideas for design of a MSVQ for operation over a time-varying channel with an informed receiver. In contrast with the discussion in section 4, here the statistics of the time-varying channel (probability distribution of the noise level) is known and hence can be used to design an improved overall system. A naive approach to tackle this problem is to provide channel protection for the worst case noise conditions and perform source coding at the resulting bit rate. The drawback of this approach is that the received signal has the same quality irrespective of the channel conditions. In fact this approach is sub optimal for all but the maxmin performance measure. An alternate approach to this problem is to use a successive approximation source coder and provide unequal error protection to the different bit streams (of varying importance) that it produces. Here we restrict the discussion to a MSVQ based successive approximation source coder. Unequal error protection can be provided for the different stages using either TEA or other more elaborate techniques e.g. [9], with the heaviest protection for the first stage and least for last (N th) stage. When the channel is clean the receiver decodes all the stages and hence achieving the best possible quality of reproduced signal. However, as the channel gets noiser, the receiver decodes only those stages that can be reliably decoded at the current level of channel noise. This results in a graceful degradation of the received signal quality.

Specifically we are interested in the problem of designing a MSVQ matched to a given unequal error protection (UEP) scheme when the statistics of the channel noise are known. Using the knowledge about the noise statistics and the UEP scheme we can estimate the set of decoder probabilities $\{P_1, ..., P_N\}$, where P_i denotes the probability

that the receiver will stop decoding at the i th stage. Our problem is now reduced to : Given a set of decoder probabilities $\{P_1, ..., P_M\}$ how do we design a MSVQ that makes use of this information.

As a first step, we introduce a modified M-L algorithm to search the MSVQ codebooks using the information provided by the decoder probabilities $\{P_i\}$.

Modifed M-L search: Ideally we would like to choose the code vectors $\{y_1, y_2, .., y_M\}$ such that the following cost is minimized,

$$D = \sum_{i=1}^{M} P_i ||x - \sum_{j=1}^{i} y_j||^2. \tag{8}$$

However, since we proceed serially from the first stage to the M th stage, at stage i we do not have information about which of the vectors will be selected in the subsequent stages. We adopt a suboptimal strategy where we replace all the subsequent stage codevectors by 0. Hence, the cost function that we seek to minimize at the I th stage is

$$D_I = \sum_{i=1}^{I-1} P_i ||x - \sum_{j=1}^{i} y_j||^2 + \left\{ \sum_{i=I}^{N} P_i \right\} ||x - \sum_{j=1}^{I} y_j||^2. \tag{9}$$

We use this cost function and perform M-L search for the best combination of codevectors from the M code books $\{C_1, C_2, ..., C_M\}$ using the procedure described in section 2.

To design the MSVQ we use this modified M-L search algorithm in a iterative design procedure similar to the one described in [7], wherein we also modify the centroid rule to account for the decoder probabilities. The results of the design applied to a Gauss-Markov process are given in Table 2. We list the values of SNR_i, the SNR if the decoding is stopped at the i th stage, for various sets of decoding probabilities $\{P_i\}$. It is interesting to observe that the design achieves significantly different values of SNR_i depending on the set of decoding probabilities. For case 1, when we are interested in the final quantization error (the case when we perform the standard M-L search based design and encoding), the value of SNR_4 is improved at the cost of drop in performance for the remaining stages. For case 2 however, when all the decoding probabilities of all the stages are equal, we have improved performance of the other stages at the cost of drop in SNR_4. Cases 4 and 5 demonstrate how the performance of two stages with high decoding probabilities can be improved at the cost of decrease in performance of the stages that have small decoding probabilities. These preliminary results clearly indicate the promise of the approach. A more detailed discussion of these issues will be presented in an upcoming publication.

References

[1] E. Bedrosian, "Weighted PCM," *IRE Trans. on Information Theory*, IT-4, March 1958, pp. 45-49.

[2] W. C. Wong, R. Steele and C. W. Sundberg, *Source-Matched Mobile Communications*. London : Pentech Press, 1995.

#	P_1	P_2	P_3	P_4	SNR_1	SNR_2	SNR_3	SNR_4
1	0	0	0	1.0	7.22	14.65	22.27	32.03
2	0.25	0.25	0.25	0.25	8.38	15.75	22.93	30.00
3	0.8	0.1	0.05	0.05	8.44	15.68	22.67	29.82
4	0	0.5	0	0.5	7.26	16.57	22.23	31.38
5	0.5	0	0	0.5	8.46	14.38	21.67	31.63

Table 2: Here we list the SNR at each stage for different probability distributions of decoding the various stages. A Gauss-Markov source (correlation 0.5) was quantized using a 4-stage MSVQ, with 64 codevectors in each stage.

[3] N. Farvardin, "A study of vector quantization for noisy channels", *IEEE Trans. on Inform. Theory* , IT-36, July 1990, pp. 799-809.

[4] K. Zeger and A. Gersho, "Pseudo-gray coding", *IEEE Trans. on Comm.* vol. 38, Dec. 1990, pp. 2147-58.

[5] D. Miller and K. Rose, "Combined Source-channel Vector Quantization Using Deterministic Annealing", *IEEE Trans. on Communications*, vol. 42, pp. 347-56, Feb 94.

[6] I. Kozintsev and K. Ramchandran, "Multiresolution Joint Source-Channel Coding using Embedded Constellations for Power Constrained Time Varying Channels", *ICASSP-96*, pp. 2343-2346.

[7] W. LeBlanc, B. Bhattacharya, S. A. Mahmoud and V. Cuperman,"Efficient Search and Design Procedures for Robust Multi-Stage VQ of LPC Parameters for 4 kb/s Speech Coding", *IEEE Trans. on Speech and Audio Proc.*, vol. 1, no. 4, Oct. 1993, pp. 373-385.

[8] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Boston: Kluwer Academic Publishers, 1992.

[9] S. Gadkari and K. Rose, "Embedded Codes for Combined Source-Channel Coding", To appear Proc. of Allerton Conference on Commun., control and computing 1996.

[10] Y. Linde, A. Buzo and R. M. Gray, "An algorithm for vector quantizer design", *IEEE Trans. on Comm.*, vol. COM-28, pp. 84-95, 1980.

AN AUTOMATIC REPEAT REQUEST SCHEME FOR JOINT LOSSLESS-SOURCE AND CHANNEL CODES

George F. Elmasry
UPS Information Services
340 Macarthur Blvd., B-335
Mahwah, NJ 07430
E-mail: GeorgeLisa@aol.com

ABSTRACT

Lossless data compression algorithms are susceptible to errors. A single error in the compressed image causes the reconstructed data to be garbled from where the error occurred to the end of the file. The error intolerance of these compression algorithms can be utilized to detect and correct errors caused by a Binary Symmetric Channel (BSC) by using a joint source and channel coding approach that introduces the redundancy needed for detection and correction before data compression. A joint automatic repeat request (ARQ) scheme and a hybrid-joint ARQ scheme are studied in this work. The introduced schemes are suitable for transmitting long files over channels with low bit error rates.

1 Introduction

In previous work [1] an approach was introduced that uses the error intolerance of lossless data compression schemes to detect and correct channel errors. With this approach redundancy is introduced in the form of a marker in the data before it is compressed by the source encoder. Three strategies for adding the redundancy marker were introduced and their performances compared using English text files and arithmetic encoding as the source encoder. These three strategies are the *particular character* (where the marker could be any character of the alphabet), the *previous character* (where the marker is the last character of the marked block), and the *average marker* (where the marker is the character that represents the average ASCII value of the marked block). The decoder examines the reconstructed data for the existence of errors. If a marker does not appear in its proper location, one or more erroneous bits exist within a certain length in the received compressed image. It was shown that this approach is capable of detecting all errors introduced in the channel regardless of the error pattern. An ARQ scheme for error detection and correction was introduced based on this scheme. In this work the approach introduced in [1], which will be referred to as joint automatic repeat request (JARQ), and hybrid JARQ will be compared with conventional ARQ schemes by evaluating the throughput efficiency of this approach against that of conventional ARQ. The conditions under which go-back-N JARQ and hybrid JARQ perform better than conventional go-back-N ARQ and conventional hybrid ARQ will be studied. The critical error rate under which hybrid JARQ performs better than hybrid ARQ will also be analyzed.

2 Throughput Efficiency for JARQ

Conventional ARQ schemes include the stop-and-wait, the go-back-N, and the selective-repeat. In this section, the throughput efficiency of go-back-N JARQ will be compared with that of go-back-N ARQ.

The throughput efficiency of go-back-N conventional ARQ is given by

$$\eta_{GBN} = \left(\frac{k}{n}\right)\left(\frac{P}{P+(1-P)N}\right), \tag{1}$$

where $\left(\frac{k}{n}\right)$ is the code rate, N (which depends on the round trip delay time) is the number of code vectors retransmitted when an error is detected, and P is the probability that a received vector will be accepted by the receiver and is given by

$$P = (1-p)^n, \tag{2}$$

where p is the BSC transition probability and n is the code vector length.

For JARQ, go-back-N ARQ will be achieved by having a receiving buffer of length M characters at the receiver, as shown in Figure 1. When the detection decoder detects an error, a NAK flag will be sent to the transmitter and the contents of the buffer will be flushed. The transmitter will go back $\overline{N}+M$ (where $\overline{N}$ is the number of characters retransmitted when an error is detected) and start re-sending again. The detection decoder is capable of detecting errors introduced in the channel by decompressing the incoming string and looking for the markers introduced in the data before compression. The relationship between the buffer length M, the strategy used to insert the marker, and the length of the marked block was studied in [1]. Figure 2 shows the probability distribution function (PDF) of detecting the errors as a function of the location over the receiver's buffer for the *previous character strategy* when the length of the marked block is 10 characters. This PDF is obtained by simulation for regular English text files using arithmetic encoding as the source encoder.

The throughput efficiency of JARQ will be given by

$$\eta_{GBNJ} = \left(\frac{1}{\triangle+1}\right)\left(\frac{\overline{P}}{\overline{P}+(1-\overline{P})(\overline{N}+M)}\right), \tag{3}$$

where $\triangle$ is the increase of the compressed data due to the added marker. $\triangle$ is obtained in [1] for each of the three strategies. For the *previous character strategy*, when the length of the marked block is 10 characters, $\triangle$ is 0.1. $\left(\frac{1}{\triangle+1}\right)$ is the code rate and it resembles $\left(\frac{k}{n}\right)$ for conventional ARQ. $\overline{P}$ is the probability that the received character is not in error and it is given by $\overline{P} = (1-p)^8$.

Figure 3, curve 1, shows $\left(\frac{P}{P+(1-P)N}\right)$ as a function of p when N=128 and n=524, while curve 2 shows $\left(\frac{\overline{P}}{\overline{P}+(1-\overline{P})(\overline{N}+M)}\right)$ as a function of p when M=64 for the same type of channel. One can see that if the difference in the code rate is not taken into consideration, the efficiency of go-back-N JARQ is less than that of conventional go-back-N ARQ.

Considering the difference in code rates, notice that to avoid a catastrophic error with conventional ARQ, a lossless source code may be avoided, or adding more parity bits to the compressed

image may be implemented, which drops the code rate $\left(\frac{k}{n}\right)$. In other words, η_{GBNJ} has a gain of R_t over η_{GBN}. If the data used by conventional ARQ is not compressed R_t is given by

$$R_t = \frac{R(1/(\triangle + 1))}{(k/n)}, \tag{4}$$

where R is the compression ratio obtained by the source encoder. If the data used by conventional ARQ is compressed R_t will be given by

$$R_t = \frac{1/(\triangle + 1)}{(k/n)}. \tag{4-a}$$

From equations 4 and 4-a, as will be explained, a gain in the efficiency could be achieved due to the fact that if the lossless source encoder is avoided in conventional ARQ, R_t could be more than one because of the factor R. On the other hand, if the data used with conventional ARQ is compressed, the gain can be achieved from the drop in the code rate $\left(\frac{k}{n}\right)$.

First, if the data used with conventional ARQ is compressed, k/n must be chosen such that the probability of catastrophic error is very low. In the comparison, k/n will be chosen such that the probability of catastrophic error $P(E)$ is upper-bounded by 10^{-20} [2], considering that go-back-N JARQ guarantees the detection of all error patterns [1]. Second, when the file used for conventional go-back-N ARQ is not compressed, k/n will be chosen such that $P(E)$ is upper-bounded by 10^{-6}. $R = 3/2$ is selected considering that for a typical English text file, R reaches the range of 2.

The curves in Figures 4 and 5 demonstrate the effect of the code rate on the throughput efficiency for the two cases. In Figure 4-a, for conventional ARQ the vector length n is 128 bits, and N is 128 vectors. Using JARQ for the same type of channel, $\overline{N}$ will be 2048 characters. The receiver's buffer length $M = 32$ characters. The curve marked (1) is conventional ARQ when k is chosen such that $P(E)$ is upper-bounded by 10^{-20}. The curve marked (2) is joint source and channel coding for the *previous character strategy* when the length of the marked block is 10 characters. Figure 4-b repeats the same plot when the vector length n is 524 instead of 128 for conventional ARQ. Notice that $\left(\frac{k}{n}\right)$ increases for longer n under the same constraint of $P(E)$, which makes the gain achieved with JARQ smaller. For such a channel, the length of the marked block is 20 characters and $M = 64$ [1]. As n increases, the performance of go-back-N ARQ approaches that of go-back-N JARQ.

Figures 5-a and 5-b show the second case (when uncompressed data is used with conventional ARQ) for the same channels. In this case k is chosen such that $P(E)$ is upper-bounded by 10^{-6} and $R = 3/2$. Obviously, the gain in the code rate obtained due to compression is considerable.

3 Throughput Efficiency for Hybrid JARQ

As explained in [1], hybrid ARQ can be achieved by correcting a high percentage of errors introduced in the channel. This is done by sending a STOP flag to the transmitter when an error is detected. The receiver's buffer will be extended to receive the string of data coming during the one-way trip delay plus the part where the error exists. The receiver then toggles bit by bit the part of the receiving buffer containing the error, and looks for the markers in the reconstructed data. (Refer to the PDF shown in Figure 2 for the average number of bits to be toggled.) This

will cause the channel to be idle until the decoder decides to send a START flag or a REPEAT flag. In this section the throughput efficiency using such an approach will be calculated.

For each character entering the buffer from the channel, there are three possibilities:

a) The buffer is error free, and the received character is not in error with probability P_c.

$$P_c = 1 - P_e, \tag{5}$$

where P_e is the probability that the received character contains an error, and P_c is given by

$$P_c = (1-p)^8. \tag{6}$$

b) The buffer contains one erroneous character with a single bit in error. This error will be corrected by the decoder with probability P_{cr}.

c) The buffer contains one or more erroneous characters that will not be corrected by the decoder with probability P_{mcr}. P_{mcr} is discussed in [1], and it depends on the strategy used for adding the marker character. Notice that $P_{cr} = 1 - P_{mcr}$, and for a low error-rate channel and a high percentage of correction $P_c \gg P_e$ and $P_{cr} \gg P_{mcr}$.

Let us calculate the throughput efficiencies for each case, and the overall throughput efficiency.

3.1 The Case of the Buffer Containing No Error

In this case, the channel is not idle and no retransmission is required. For the character entering the buffer to be accepted, the average number of retransmissions is

$$T_a = P_c + 2P_c(1-P_c) + 3P_c(1-P_c)^2 + \ldots\ldots = \frac{1}{P_c},$$

and the throughput efficiency is

$$\eta_a = \frac{1}{T_a}\left(\frac{1}{\triangle + 1}\right) = \left(\frac{1}{\triangle + 1}\right) P_c. \tag{7}$$

3.2 The Case of the Buffer Containing an Error (to be Corrected)

When the decoder detects an error, a STOP flag will be sent to the transmitter. Until the decoder decides to send a START flag, there will be idle time D_1, which is the average time needed by the decoder to correct the error. If the channel signaling rate is τ, the transmitter could transmit $D_1\tau$ characters during the idle time. The average number of characters that can be accepted by the decoder T_b is

$$T_b = D_1\tau.P_c + 2D_1\tau.P_c(1-P_c) + 3D_1\tau.P_c(1-P_c)^2\ldots\ldots = \frac{D_1\tau}{P_c},$$

and hence the throughput efficiency will be

$$\eta_b = \left(\frac{1}{\triangle + 1}\right)\left(\frac{P_c}{D_1\tau}\right). \tag{8}$$

3.3 The Case of the Buffer Containing an Error (Not to be Corrected)

In this case, a STOP flag will be sent to the transmitter, and until the decoder sends a REPEAT flag, the transmitter can send $D_2\tau$ characters, where D_2 is the average time needed by the decoder to decide that the error could not be corrected. Also, by taking into consideration the N characters that will be retransmitted, the average number of characters that could be accepted by the decoder is

$$T_c = T_1 + T_2,$$

where

$$T_1 = D_2\tau.P_c + 2D_2\tau.P_c(1-P_c) + 3D_2\tau.P_c(1-P_c)^2 ..... = \frac{D_2\tau}{P_c},$$

and

$$T_2 = 1.P_c + (N+1)P_c(1-P_c) + (2N+1)P_c(1-P_c)^2 + ...$$

$$T_2 = 1 + \frac{N(1-P_c)}{P_c} = \frac{P_c + (1-P_c)N}{P_c},$$

and hence the throughput efficiency is

$$\eta_c = \left(\frac{1}{\triangle + 1}\right) \cdot \left(\frac{P_c}{N + D_2\tau + P_c(1-N)}\right). \tag{9}$$

3.4 Overall Efficiency

From equations (7), (8), and (9), we can calculate the throughput efficiency of hybrid JARQ as follows:

$$\eta_{JH} = \eta_a.(P_c)^M + \eta_b.P_{cr}.\left(1-(P_c)^M\right) + \eta_c.P_{mcr}.\left(1-(P_c)^M\right), \tag{10}$$

where P_c is given in equations (5) and (6), M is the receiver's buffer length, and $(P_c)^M$ is the probability that it contains no error, hence the throughput efficiency of hybrid JARQ becomes

$$\eta_{JH} = \left(\frac{1}{\triangle + 1}\right)(1-P_e)\left\{(P_c)^M + \left(1-(P_c)^M\right)\left(\frac{1-P_{mcr}}{D_1\tau}\right)\right.$$
$$\left. + \left(1-(P_c)^M\right)\left(\frac{P_{mcr}}{N + D_2\tau + P_c(1-N)}\right)\right\}. \tag{11}$$

From equation (11), we can notice the following:

- $\left\{(P_c)^M + \left(1-(P_c)^M\right)\left(\frac{1-P_{mcr}}{D_1\tau}\right) + \left(1-(P_c)^M\right)\left(\frac{P_{mcr}}{N+D_2\tau+P_c(1-N)}\right)\right\}$ is always less than 1, and it approaches 1 as P_e approaches zero.

- The smaller the N, $D_1\tau$, and $D_2\tau$, the higher the efficiency.

- As P_{mcr} approaches zero, η becomes

$$\eta_{JH} = \left(\frac{1}{\triangle + 1}\right)(1-P_e)\left[(P_c)^M + \left(1-(P_c)^M\right)\left(\frac{1}{D_1\tau}\right)\right]; \tag{12}$$

as $D_1\tau$ approaches 1 (no idle time), η becomes

$$\eta_{JH} = \left(\frac{1}{\triangle + 1}\right)(1-P_e); \tag{13}$$

and as P_e approaches zero, η approaches the code rate,

$$\eta_{JH} = \left(\frac{1}{\triangle + 1}\right). \tag{14}$$

4 The Limit Probability of Error

One can see that the hybrid JARQ scheme presented in section 3 does not require extra redundancy for correction since the correction capability is achieved with the same redundancy added for detection. This is not the case for conventional hybrid ARQ, where extra parity bits are needed for correction and hence the code rate drops. The cost of this correction capability is delay time, during which the channel will be idle until the receiver sends either a REPEAT or a RESEND flag. With today's technology this delay time could be minimized.

Studies of conventional ARQ [2] showed that selective-repeat ARQ gives the best performance of all three ARQ schemes mentioned in section 2. These studies also showed that selective-repeat hybrid ARQ schemes give less throughput efficiency than selective-repeat ARQ at low error rates because of the extra parity bits added for correction. This performance is reversed at high error rates, where the selective-repeat ARQ throughput efficiency drops sharply [2]. Since our present interest is low error-rate channels, our comparison will be limited to this range. Hybrid JARQ will be compared with ideal selective-repeat ARQ under the same condition used for comparison in section 2.

The throughput efficiency of hybrid JARQ is given by equation (11). The throughput efficiency of the ideal selective-repeat ARQ is given by

$$\eta_{SR} = \left(\frac{k}{n}\right)(1 - P_e). \tag{15}$$

As shown in section 2, if the file used with conventional ARQ is uncompressed, the gain achieved with compression will be considerable. Therefore, we will consider the case when the file used for conventional ARQ is compressed. Figure 6, curve 2, shows the throughput efficiency given in equation (15) of ideal selective-repeat ARQ when $P(E)$ is upper-bounded by 10^{-20}. Curve 1 shows the throughput efficiency of hybrid JARQ given in equation (11) when $D_1\tau$ is 10^6 bits, $D_2\tau$ is $3 * 10^6$ bits, and P_{mcr} is 0.01. It is clear that as the error rate increases, the performance of hybrid JARQ drops. From the curves in Figure 6, we can see that for low error rates (where ideal selective-repeat ARQ performs better than hybrid selective-repeat ARQ), hybrid JARQ performs better than both hybrid selective-repeat and ideal selective-repeat ARQ's.

The limit probability of error given by the intersection of the two curves in Figure 6 is decided by the overall compression ratio given in equation (4). From equations (11) and (15), to have a higher throughput efficiency using hybrid JARQ, we should have

$$\left(\frac{1}{\triangle + 1}\right)(1-p)^8 \left\{(1-p)^{8M} + \left(1 - (1-p)^{8M}\right)\left(\frac{1 - P_{mcr}}{D_1\tau}\right)\right.$$
$$\left. + \left(1 - (1-p)^{8M}\right)\left(\frac{P_{mcr}}{M + D_2\tau + (1-p)^8(1-M)}\right)\right\} R \geq \left(\frac{k}{n}\right)(1-p)^n. \tag{16}$$

Substituting the value of R_t from equation (4) gives

$$(1-p)^{-n+8(M+1)} + \left((1-p)^{-n+8} - (1-p)^{-n+8(M+1)}\right)\left(\frac{1-P_{mcr}}{D_1\tau}\right)$$

$$+\left((1-p)^{-n+8} - (1-p)^{-n+8(M+1)}\right)\left(\frac{P_{mcr}}{M+D_2\tau+(1-p)^8(1-M)}\right) \geq 1/R_t. \qquad (17)$$

In equation (17), when the length of the code vector n used in conventional ARQ is more than $8(M+1)$, the inequality will be satisfied for any value of p, since the term $(1-p)^{-n+8(M+1)}$ will be more than 1, given that $(1/R_t) \prec 1$ (i.e., $R_t \succ 1$). The curve in Figure 7 shows the critical error rate as a function of the overall compression ratio R_t in this case.

5 Conclusion

In this work a joint lossless-source and channel coding ARQ approach that utilizes the data compression algorithm's susceptibility to errors to detect and correct errors was presented. A go-back-N JARQ scheme was discussed as was the condition under which it performs better than go-back-N ARQ. Also shown was a way to adapt a hybrid JARQ scheme without having to add more redundancy for correction. That JARQ is suitable for channels with low bit error rates was illustrated, and the limit error rate under which hybrid JARQ performs better than conventional ideal selective-repeat ARQ and conventional hybrid ARQ was shown.

In the future, the intent is to compare the throughput efficiencies of the three JARQ strategies at different error rates, and find the optimum performance for a given channel. For hybrid JARQ, the effect of the idle time needed by the receiver will be studied. The relationship between the correction capability and the efficiency will be analyzed.

6 References

[1] G.F. Elmasry and Y. Bar-Ness, "An Automatic Repeat Request Scheme for Error Detection and Correction of Compressed Data," *Proc. of CISS'96*, Princeton, NJ, March 1996, pp. 561-566.

[2] Shu Lin and Daniel J. Costello, Jr., *Error Control Coding: Fundamentals and Applications.* Englewood, NJ: Prentice-Hall, Inc., 1985.

[3] Glen G. Langdon, Jr., "An Introduction to Arithmetic Coding," *IBM Journal of Research and Development,* Vol. 28, No. 2, March 1984.

[4] Y. Bar-Ness and P. Narasimhan, "Error Propagation and Error Correction for Universal Coding Systems," *International Symposium on Information Theory and Applications,* Waikiki, HI, November 1990.

[5] Richard E. Blahut, *Principles and Practice of Information Theory.* Reading, MA: Addison-Wesley Publishing Co., 1990.

[6] C.B. Jones, "An Efficient Coding System for Long Source Sequences," *IEEE Transactions on Information Theory,* Vol. IT-27, No. 3, May 1981, pp. 280-291.

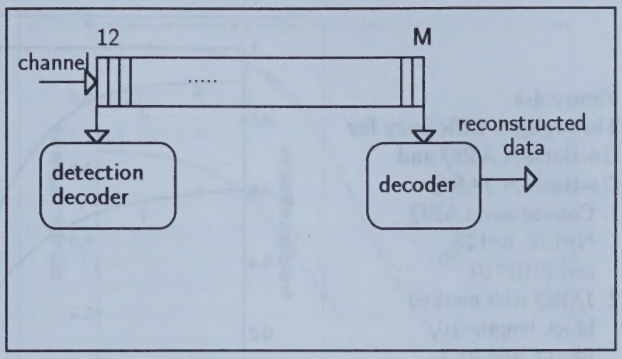


Figure 1
The Receiver Buffer

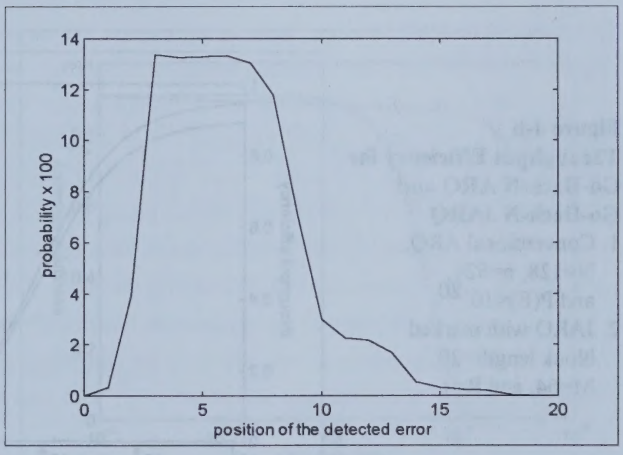


Figure 2
The PDF of Errors over the Receiver Buffer

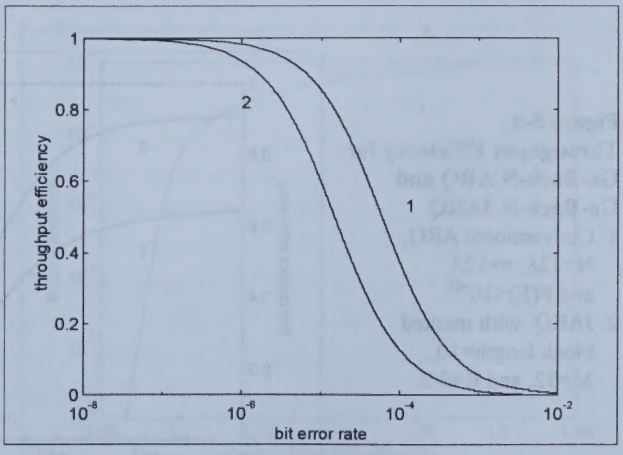


Figure 3
Throughput Efficiency for Go-Back-N ARQ and Go-Back-N JARQ When the Code Rate Is Not Taken into Consideration

1. Conventional ARQ, N=128 and n=524.
2. JARQ with M=64.

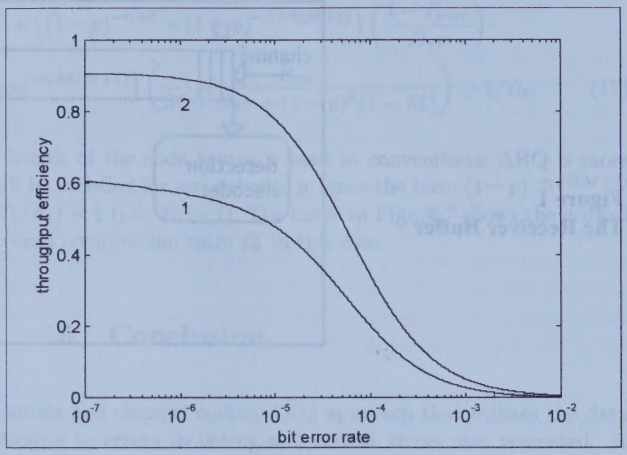


Figure 4-a
Throughput Efficiency for Go-Back-N ARQ and Go-Back-N JARQ
1. Conventional ARQ, N=128, n=128, and $P(E)<10^{-20}$.
2. JARQ with marked block length=10, M=32, and R=1.

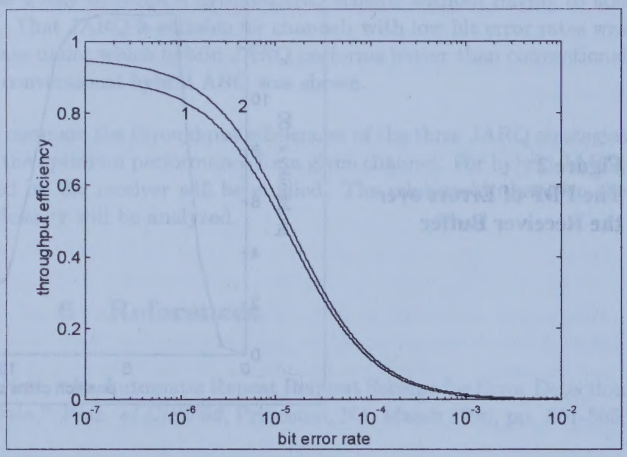


Figure 4-b
Throughput Efficiency for Go-Back-N ARQ and Go-Back-N JARQ
1. Conventional ARQ, N=128, n=524, and $P(E)<10^{-20}$.
2. JARQ with marked block length=20, M=64, and R=1.

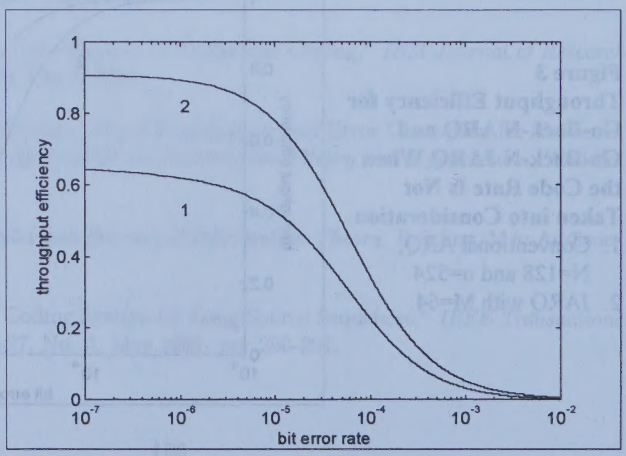


Figure 5-a
Throughput Efficiency for Go-Back-N ARQ and Go-Back-N JARQ
1. Conventional ARQ, N=128, n=128, and $P(E)<10^{-6}$.
2. JARQ with marked block length=10, M=32, and R=3/2.

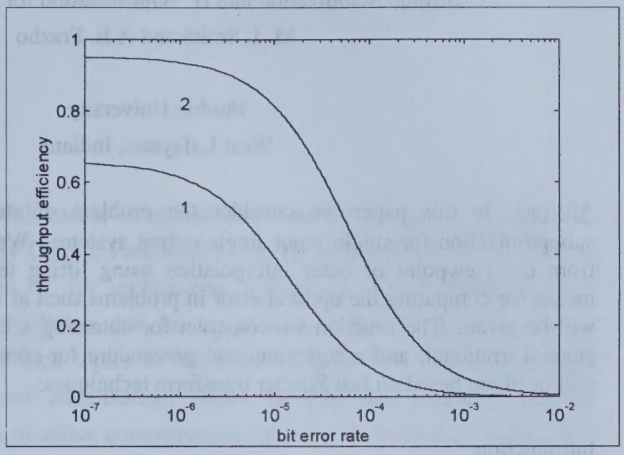


Figure 5-b
Throughput Efficiency for Go-Back-N ARQ and Go-Back-N JARQ
1. Conventional ARQ, N=128, n=524, and $P(E)<10^{-6}$.
2. JARQ with marked block length=20, M=64, and R=3/2.

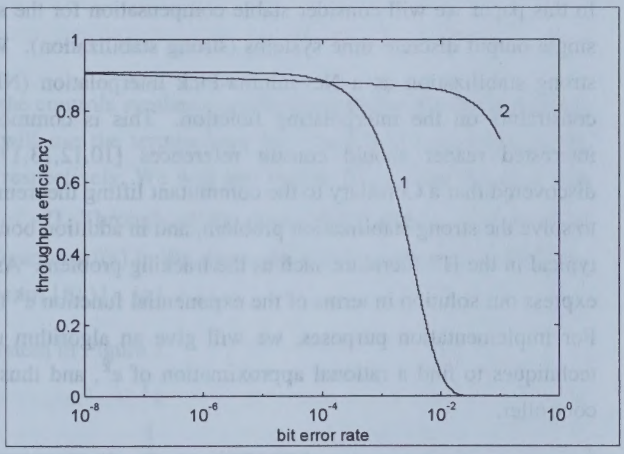


Figure 6
Throughput Efficiency of Hybrid-JARQ and Hybrid ARQ
1. Hybrid-joint source and channel coding, marked block length=10.
2. Conventional ideal selective-repeat ARQ, n=524 and $P(E)<10^{-20}$

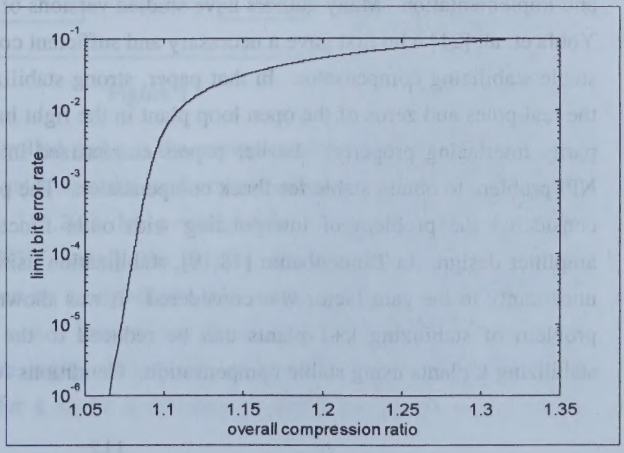


Figure 7
The Limit Bit Error Rate as a Function of the Overall Compression Ratio
n=524, $P(E)<10^{-20}$, and M=64.

Strong Stabilization and H^∞ Optimization for SISO Systems

M. J. Smith and A.E. Frazho

Purdue University
West Lafayette, Indiana

Abstract In this paper we consider the problem of strong stabilization with H^∞ suboptimization for single input single output systems. We will approach this problem from the viewpoint of outer interpolation using lifting techniques. A computational means for computing the optimal error in problems such as tracking with outer functions will be given. The function we construct for obtaining a bound on the error will be in general irrational, and a computational proceedure for computing a rational interpolant will be given based on fast Fourier transform techniques.

Introduction

In this paper we will consider stable compensation for the stabilization of single input - single output discrete time systems (strong stabilization). We will view the problem of strong stabilization as a Nevanlinna-Pick interpolation (NPI) problem with additional constraints on the interpolating function. This is common in the literature, and the interested reader should consult references [10,12,13,17,19] for details. We have discovered that a Corollary to the commutant lifting theorem [22] can be used effectively to solve the strong stabilization problem, and in addition bound the H^∞ norm of problems typical in the H^∞ literature such as the tracking problem. As in Vidyasagar [12], we will express our solution in terms of the exponential function e^k for k analytic in the unit disc. For implementation purposes, we will give an algorithm using fast Fourier transform techniques to find a rational approximation of e^k, and thus a rational stable stabilizing controller.

There are many advantages to the design of stable compensators, particularly in testing and implementation. Many authors have studied versions of this problem beginning with Youla et. al. [21] who first gave a necessary and sufficient condition for the existence of a stable stabilizing compensator. In that paper, strong stabilization was given in terms of the real poles and zeros of the open loop plant in the right half complex plane (called the parity interlacing property). Earlier papers emphasized interpolation constraints in the NPI problem to obtain stable feedback compensation. The paper by Ball and Helton [17] considered the problem of interpolating with outer functions with an application to amplifier design. In Tannenbaum [18,19], stabilization using a stable compensator with uncertainty in the gain factor was considered. It was shown in Vidyasagar [20] that the problem of stabilizing k+1 plants can be reduced to the problem of simultaneously stabilizing k plants using stable compensation. Conditions for optimality in the H^∞ norm

for multivariable systems with stable compensation was developed in Sideris and Safonov [16]. In the paper by Dorato, Park, and Lee [11], an algorithm was developed for interpolation with units in H^∞ for applications to feedback stabilization, and other results concerning unit interpolation can be found in [10,11,12]. Several authors [1,7,8,13,14,15] have studied such topics as the minimal order of a stable stabilizing compensator and simultaneous stabilization. More recently, the papers [2,3,4,5,6] consider multivariable design of stable compensators for H^2 and simultaneous $H^2 - H^\infty$ suboptimal problems using Riccati techniques.

1. Problem Formulation.

In this section we will set up the controls synthesis problem and define the corresponding mathematical problem. We will use the terminology Im(·) and Ker(·) to represent the image and kernal of a space, respectively. We will also denote $\mathcal{D}$ to mean the open unit disc and $\overline{\mathcal{D}}$ for the closure of $\mathcal{D}$. Throughout this paper $\mathbb{RH}^\infty$ defines the set of all uniformly bounded analytic functions f(z) in the open unit disc such that $f(z) = f^*(\overline{z})$ for all $|z|<1$. Moreover, $\|f\|_\infty = \sup\{|f(z)| : |z| < 1\}$.

Now consider the feedback system in Figure 1.

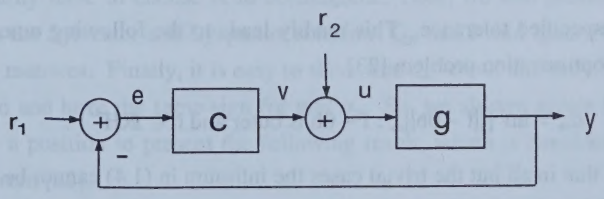


Figure 1

Here, g is the plant to be controlled and c is the compensator in $\mathbb{RH}^\infty$ to be synthesized that stabilizes the system and satisfies some performance criteria. By consulting Youla's original paper [21] (see also [16,20]), let $g = n/d$ where n and d are rational $\mathbb{RH}^\infty$ functions, let p and q in $\mathbb{RH}^\infty$ satisfy the Bezout identity $dq + np = 1$. Then all stabilizing controllers are given by $c = (p+dh)/(q-nh)$ where h is any function in $\mathbb{RH}^\infty$. From this characterization of c, it is known (see Doyle, et,al [9]) that $p + dh$ and $q - nh$ have no common zeros in the region of required analyticity. Thus, forcing the term $q - nh$ to be outer suffices for c to be a function in $\mathbb{RH}^\infty$ and hence c is a stable

stabilizing compensator. Now keeping this fact in mind, consider the suboptimization problem of finding a c in $\mathbb{RH}^\infty$ that internally stabilizes the closed loop system in Figure 1 and for some predetermined tolerance $\gamma > 0$,

$$\|(1 + gc)^{-1}\|_\infty < \gamma \quad \text{for } c \in \mathbb{RH}^\infty . \tag{1.1}$$

From Figure 1, this is essentially a tracking problem. By using the parameterization of all stabilizing controllers, $(1+gc)^{-1} = d(q-nh)$. By requiring $q - nh$ to be outer, we obtain the equivalent problem in terms of the Bezout identity

$$\|d(q - nh)\|_\infty < \gamma \quad (\text{for } h \in H^\infty \text{ and } q - nh \text{ outer}) . \tag{1.2}$$

Therefore, the problem of finding a stable compensator c such that the tracking error $\|(1 + gc)^{-1}\|_\infty \le \gamma$ is equivalent to finding a function $h \in \mathbb{RH}^\infty$ satisfying (1.2). In this case, the corresponding compensator c is given by $c = (p+dh)/(q-nh)$.

Now consider the inner outer factorization $\theta\, \theta_{out}$ of dn and set $f = dq$, then (1.2) is equivalent to the following $\mathbb{RH}^\infty$ problem. Given an f in $\mathbb{RH}^\infty$ and an inner function θ in $\mathbb{RH}^\infty$, find an h in $\mathbb{RH}^\infty$ satisfying

$$\|f - \theta h\|_\infty \le \gamma \text{ and } f - \theta h \text{ is outer } (h \in \mathbb{RH}^\infty) . \tag{1.3}$$

where γ is a prespecified tolerance. This readily leads to the following outer version of the Sarason H^∞ optimization problem [23],

$$d_\infty = \inf \{\|f - \theta h\|_\infty : f - \theta h \text{ is outer and } h \in \mathbb{RH}^\infty\} . \tag{1.4}$$

It is well known that in all but the trivial cases the infimum in (1.4) cannot be replaced by the minimum [2]. That is, the optimum produces a singular inner function. This observation also follows from the commutant lifting theorem.

There has been an immense amount of literature published on the subject of H^∞ interpolation problems with units in $\mathbb{RH}^\infty$. In this note, we have two observations. First we have constructed explicitly some formulas for computing the optimal value d_∞ in (1.4). Secondly, by applying a corollary to the commutant lifting theorem [22] along with fast Fourier transforms and a balanced version of the Kalman-Ho algorithm, we arrive at a rational h which approximates solutions to the H^∞ problem in (1.4) as close as desired.

2. State Space Solutions

In this section we will develop state space solutions to the optimization problem proposed in (1.3). For simplicity of presentation, let us assume that θ is a finite dimensional Blaschke product with distinct zeros $\{\alpha_1,\alpha_2,...,\alpha_m\}$. Now let $(\cdot)^T$ denote the transpose and let A, B, and C_n be the operators defined by

$$A = \text{diag}(\alpha_1,\alpha_2,...\alpha_m), \quad B = [1,1,...1]^T \text{ and}$$
$$C_n = [\ln(f(\alpha_1)) + 2\pi n_1 i,\ \ln(f(\alpha_2)) + 2\pi n_2 i,...,\ln(f_m(\alpha_m) + 2\pi n_m i]^T \tag{2.1}$$

where the n_i's are integers in $\mathcal{B}$. Here, $\mathcal{B}$ is the set of integers satisfying $n_j = 0$ if α_j is real, otherwise $n_j = -n_q$ if $\alpha_j = \bar{\alpha}_q$. Let P be the controllability grammian for the pair $\{A,B\}$ which is the solution to the following Lyapunov equation

$$P = APA^* + BB^* . \tag{2.2}$$

Also let Q_n be the solution to the following Sylvester equation

$$Q_n = AQ_nA^* + C_nB^* . \tag{2.3}$$

Because A is a diagonal matrix, both P and Q_n are Pick type matrices. However, we do not necessarily have to choose A to be diagonal. Thus, we will present our algorithms based upon the Sylvester and Lyapunov solutions Q_n and P and ignore the special forms of the Pick matrices. Finally, it is easy to show that $d_\infty < \infty$ if and only if all the $f(\alpha_j)$ are all non zero and have the same sign for real α_j. So we always assure that $d_\infty < \infty$. We are now in a position to present the following result, which is based on the commutant lifting theorem [22].

Theorem 2.1 *Let* A, B, *and* C_n *be the operators defined in* (2.1) *where the* n_j*'s are integers in* $\mathcal{B}$. *Let* P *and* Q_n *be the solutions to the Lyapunov and Sylvester equations in* (2.2) *and* (2.3), *respectively. Then*

$$d_\infty^2 = \exp(\min_{n\in\mathcal{B}}\{\lambda_{\max}(Q_n + Q_n^*)P^{-1}\}) . \tag{2.4}$$

Notice from the relation given for Q_n in (2.3), we see that as n_j gets large, the maximum eigenvalue of $(Q_n + Q_n^*)P^{-1}$ gets large so the previous search is over a finite set. The advantage of this result is that we have only a finite parameter search over n. It's also

important to note that for any n, we obtain a bound on the infinity norm. The next section presents an algorithm for computing h in $\mathbb{RH}^\infty$ in (1.3).

3. Algorithm For Computing h

Using the previous notation, we are ready to state the following algorithm which enables us to compute an approximate solution to the $\mathbb{RH}^\infty$ optimization problem in (1.4). As noted earlier, the optimal solution is in general a singular inner function and not outer, hence we must stay away from the optimal solution by an arbitrary epsilon.

Algorithm As before, assume that the $f(\alpha_j)$ all have the same sign for real (α_j). Let γ be any real number such that $d_\infty < e^\gamma$ where d_∞ has been computed in (2.4). Let P be the solution to the Lyapunov equation in (2.2) and Q_n be the solution to the Sylvester equation in (2.3) for the optimal n in $\mathcal{B}$, then

$$d_\infty^2 = \exp\left[\lambda_{max}(Q_n + Q_n^*)P^{-1}\right]. \tag{3.1}$$

Next, set $\gamma = \left[\lambda_{max}(Q_n+Q_n^*)P^{-1}\right]/2+\delta$ for arbritrary $\delta > 0$ and compute the matrices

$$K_1 = (P^{-1}Q_n - \gamma I)\left[2\gamma I - P^{-1}(Q_n+Q_n^*)\right]^{-1}\left[P^{-1}Q_n^* - (\gamma+1)I\right] - I \tag{3.2a}$$

and

$$K_2 = \left[2\gamma I - P^{-1}(Q_n+Q_n^*)\right]^{-1}\left[P^{-1}Q_n^* - (\gamma+1)I\right] + I\,. \tag{3.2b}$$

Furthermore, let p and q be the rational functions given by

$$p = 1 + B^*(I - zA^*)^{-1}K_1P^{-1}B \quad \text{and} \quad q = 1 - B^*(I - zA^*)^{-1}K_2P^{-1}B\,. \tag{3.3}$$

Then the function $k = \gamma - p/q$ is a rational function in $\mathbb{RH}^\infty$ and exp(k) is a well defined outer function of the form $\exp(k) = f - \theta h$ for an h in H^∞. Solving for h we obtain

$$h = \frac{f - \exp(k)}{\theta} = \frac{f - \exp(\gamma - p/q)}{\theta} = \sum_{j=0}^{\infty} h_j z^j \tag{3.4}$$

where the h_j's are the Fourier coeficients of h. Using fast Fourier transform techniques, compute $\{h_j\}$. Now using a balanced Kalman-Ho type realization and excluding the insignificant singular values, we arrive at a rational approximation of h(z) given by $h(z) \approx m(z)/d(z)$ where m(z) and d(z) are polynomials. Because h(z) is analytic in $\overline{\mathcal{D}}$, the

previous approximation techniques yield an outer function $f - \theta m(z)/d(z)$ that satisfies the following inequality constraint

$$\|f - \theta m/d\|_\infty \leq \exp(\gamma) + \varepsilon \,. \tag{3.5}$$

where ε is an arbitrary value greater than zero. The following section outlines the proofs based upon the central solution to the commutant lifting theorem [22].

4.0 Proofs Via The Commutant Lifting Theorem.

In this section, we will sketch a proof of these results based on the positive real version of the commutant lifting theorem.

To begin, first notice that if $f - \theta h$ is an outer function in $\mathbb{RH}^\infty$, then there exists a function k analytic in the open unit disc satisfying $f - \theta h = \exp(k(z))$ (see Hoffman [24]). In this case, it follows that there exists a function $k_n(z)$ dependent upon the branch cut $\mathcal{B}$ that satisfies $f - \theta h = \exp(k_n - \theta h_o)$ where h_o is a function in $\mathbb{RH}^\infty$. This observation readily shows that the $\mathbb{RH}^\infty$ optimization problem in (1.4) is equivalent to

$$\begin{aligned} d_\infty &= \inf \{ \|f - \theta h\|_\infty : h \in \mathbb{RH}^\infty \text{ and } f - \theta h \text{ is outer} \} \\ &= \inf \{ \|\exp(k_n - \theta \hat{h})\|_\infty : \hat{h} \in \mathbb{RH}^\infty \text{ and } n \in \mathcal{B} \} \\ &= \inf \{ \|\exp(\operatorname{Re}(k_n - \theta \hat{h}))\|_\infty : \hat{h} \in \mathbb{RH}^\infty, n \in \mathcal{B} \} \,. \end{aligned} \tag{4.1}$$

Thus, solving the H^∞ optimization problem in (1.3) can be viewed as a special case of the following positive real interpolation problem. Given $k_n \in \mathbb{RH}^\infty$ and an inner function θ, find $d_p = \min\{ d_{p,n} : n \in \mathcal{B} \}$ where $d_{p,n}$ is the error determined by

$$d_{p,n} = \inf\{ \sup_{|z|<1} \operatorname{Re}(k_n - \theta h) : h \in \mathbb{RH}^\infty \} \,. \tag{4.2}$$

Since $\exp(x)$ is an increasing function for real x, we have $d_\infty = \exp(d_p)$. The solution to this problem can be solved by a direct application of the positive real version of the commutant lifting theorem based upon Theorem 7.2 of chapter 9 in [22]. It's proof can be obtained by using the Cayley transform in conjunction with that theorem. Now we have the following Lemma 4.1.

Lemma 4.1 *Let* k_n *be a function in* $\mathbb{RH}^\infty$ *and* θ *a rational inner function in* $\mathbb{RH}^\infty$ *and let* $\mathcal{H} = H^2 \ominus \theta H^2$. *Let* L_n *be a finite dimensional operator from* $\mathcal{H}$ *into* $\mathcal{H}$ *defined by* $L_n x = P_{\mathcal{H}} k_n x$ *where* $x \in \mathcal{H}$ *and* $P_{\mathcal{H}}$ *denotes the orthogonal projection from* H^2 *onto* $\mathcal{H}$. *Consider the optimization problem in (4.2). Then* $d_{p,n}$ *is the maximum eigenvalue of* $L_n + L_n^*$. *Moreover, if* $k = k_n - \theta h$ *is a solution to the previous optimization problem for some* $h \in \mathbb{RH}^\infty$, *then* $2\,\mathrm{re}(k) = d_{p,n}^2$ *a.e. In particular,* $\exp(k)$ *is an inner function multiplied by a constant.*

Let n be the element of the branch cut corresponding to the optimal value, $d_\infty^2 = \exp(d_p) = \exp(d_{p,n})$. Let $k_n(z)$ be any function in H^∞ satisfying $k_n(\alpha_j) = \ln(f(\alpha_j)) + 2\pi n_j i$. In particular, one such $k_n(z)$ is given by

$$k_n(z) = B^*(I - zA^*)P^{-1}C_n \tag{4.3}$$

where A, B, and C_n were previously defined in (2.1) and P is the solution to the Lyapunov equation in (2.2). Now let L_n be the operator on $\mathcal{H}$ be given by $L_n x = P_{\mathcal{H}} k_n x$ for x in $\mathcal{H}$. Then the optimal error d_∞ to the Sarason outer H^∞ optimization problem in (1.4) is given by

$$d_\infty^2 = \exp(\lambda_{\max}(L_n + L_n^*)) \,. \tag{4.4}$$

Now according to the previous Lemma 4.1, if $k = k_n - \theta h$ is the optimal solution in (4.2), then in general $\exp(k)$ is a singular inner function and hence is not outer. Now let W be the operator from $\mathcal{X}$ onto $\mathcal{H}$ defined by

$$Wx = B^*(I - zA^*)^{-1}P^{-1}x \,. \quad (x \in \mathcal{X}) \tag{4.5}$$

Then it turns out that W is an invertible operator from $\mathcal{X}$ into $\mathcal{H}$ (see [22] for details). Thus, using this similiarity transformation W we obtain

$$\lambda_{\max}(L_n + L_n^*) = \lambda_{\max}(Q_n + Q_n^*)P^{-1} \,. \tag{4.6}$$

Hence, (4.6) readily yields our formula in (3.1) for d_∞^2.

The following result can be obtained by applying Cayley transform techniques to Corollary 3.5 in [25].

Lemma 4.2 *Let* S *be the unilateral shift on* H^2 *and* $\mathcal{H}$ *an invariant subspace for* S^*. *Let* T *be the contraction on* $\mathcal{H}$ *defined by* $T = P_{\mathcal{H}} S | \mathcal{H}$. *Let* L *on* $\mathcal{H}$ *be any operator commuting with* T *and satisfying the bound* $L + L^* < 2\gamma$. *Let* p *and* q *be the functions defined by*

$$p = (I - P_{\mathcal{H}})1 + (L - \gamma)(2\gamma - L^* - L)^{-1}(L^* - \gamma - I)P_{\mathcal{H}}1 \qquad (4.7)$$
$$q = (I - P_{\mathcal{H}})1 - (2\gamma - L - L^*)^{-1}(L^* - \gamma - I)P_{\mathcal{H}}1$$

where as before, $P_{\mathcal{H}}$ *is the orthogonal projection onto* $\mathcal{H}$ *and* 1 *is the constant function in* H^2. *Then* $k = \gamma - p/q$ *is analytic in* $\overline{\mathcal{D}}$ *and* $Lx = P_{\mathcal{H}} kx$ *for* $x \in \mathcal{H}$. *Finally,* k *satisfies the bound* $2\,\mathrm{re}(k) \le \gamma$.

As before, let k_n be given by the state space representation in (4.3), and set $L_n x = P_{\mathcal{H}} k_n x$. By using the similarity transformation W on L_n along with Lemma 4.2 we can convert the formula in (4.7) to the formulas in (3.2) and (3.3). (The computations are much too involved to be given here and will be presented elsewhere.) Hence, one obtains Theorem 2.1 and the algorithm in Section 3. Finally, we must emphasize that Lemma 4.2 is general in the sense that it holds for irrational functions as well as rational functions, and thus can be used to solve for a stable stabilizing compensator for certain irrational functions.

References

[1] O. Toker, "On the Order of Simultaneous Stabilizing Compensators," IEEE TAC, Vol. 41, No. 3, March 1996.

[2] V. Kapila and W. Haddad, "H_2 and Mixed-H_2/H_∞ Stable Stabilization," Proc. of 34th CDC, December 1995.

[3] M. Peters and P. Iglesias, "The Minimum Entropy Controller as a Mixed H_2/H_∞ Controller: the Time Varying Case," Proc. of 34th CDC, December 1995.

[4] W. Haddad and D. Berstein, "Stable H_2/H_∞ Controller Synthesis," Proc. of 33th CDC, December 1994.

[5] Y. Wang and D. Bernstein, "H_2-suboptimal Stable Stabilization," *Automatica*, Vol. 30, No. 11, pp. 1797-1800, 1994.

[6] Y. Wang, W. Haddad, and D. Bernstein, "Robust Strong Stabilization via Modified Popov Controller Synthesis," IEEE TAC, Vol. 39, No. 11, November 1994.

[7] A. Gindes, "Reliable Stabilization of Linear Plants Using a Two Controller Configuration," *System and Control Letters* 23 (1994), pp. 297-304.

[8] V. Blondel, M. Gevers, R. Mortini and R. Rupp, "Simultaneous Stabilization of Three or More Plants: Condition on the Positive Real Axis Do Not Suffice," *SIAM J. Control of Optimization*, Vol. 32, No. 2, pp. 572-590, March 1994.

[9] J. C. Doyle, R. A. Francis and A. R. Tannenbaum, *Feedback Control Theory*, McMillan, New York, 1992.

[10] Y. Ohta, "Unit Interpolation in H_∞: Bounds of Norm and Degree Interpolants," *Systems of Control Letters* 17 (1991) 251-256.

[11] P. Dorato, H. Park and Y. Li, "An Algorithm for Interpolation with Units in H^∞ with Applications to Feedback Stabilization," IFAC, p. 427, 1989.

[12] S. Hare and M. Vidyasagar, "Sensitivity Minimization and Robust Stabilization by Stable Controller," MTNS, pp. 293-320, 1989.

[13] K. Wei and R. Barmish, "An Iterative Design Procedure for Simultaneous Stabilization of MIMO Systems," IFAC, pp. 643-652, July 1987.

[14] M. C. Smith and M. P. Sondergeld, "On the Order of Stable Compensation," IFAC, pp. 127-129, 1986.

[15] B. Ghosh, "Some New Results on the Simultaneous Stabilization of a family of SISO System," *System of Control Letters* 6 (1985), pp. 35-45.

[16] A. Sideris and M. G. Sefaros, "Infinity Norm Optimization with a Stable Controller," ACC, pp. 804-805, 1985.

[17] J. Ball and J. W. Helton, "Interpolation with Outer Functions and Gain Equalization in Amphifiers," MTNS '79 1979, pp. 41-49.

[18] A. Tannenbaum, "Modified N-P Interpolation and Feedback Stabilization of Linear Plants with Uncertainty in the Gain Factor," *Int. J. Control*, 1982, Vol. 36,

No. 2, pp. 231-236.

[19] A. Tannenbaum, "Feedback Stabilizing of Linear Dynamical Plants with Uncertainty in the Gain Factor," *Intl. J. Control*, 1980, Vol. 32, No. 1, pp. 1-16.

[20] M. Vidyasagor and N. Viswanadham, "Algebraic Design Techniques for Reliable Stabilization," IEEE TAC, Vol. AC-27, No. 5, October 1982.

[21] D. C. Youla, J. J. Bougiorno Jr. and C. N. Lu, "Single-Loop Feedback Stabilizing of Linear Multivariable Plants," *Automatica*, Vol. 10, pp. 159-173, 1974.

[22] C. Foias and A. E. Frazho, "The Commutant Lifting Approach to Interpolation Problems, Operator Theory: Advances & Applications," Vol. 44, Birkhäuser Verlag, Basel, 1990.

[23] D. Sarason,"Generalized Interpolating in H^∞," *Trans. American Math. Soc.*, 127 (1967), pp. 179-203.

[24] K. Hoffman, *Banach Spaces of Analytic Functions*, Englewood Cliffs, Prentice Hall, New Jersey, 1962.

[25] C. Foias, A. E. Frazho and W. S. Li, "The Exact H^2 Estimate for the Central H^∞ Interpolant," *Operator Theory; Advances of Applications*, Vol. 64, 1993, Birkhauser Verlag, Basel.

Robust Stabilization for Systems with Rank-One Uncertainty Structure*

Jianbo Lu and Mario A. Rotea
School of Aeronautics and Astronautics
Purdue University
West Lafayette, IN 47907-1282

Abstract

This paper solves a class of robust control problems, with output feedback, for systems with mixed parametric uncertainty and unmodeled dynamics. This class of problems is characterized by a special rank-one assumption on the transfer matrix of the nominal plant. Under this assumption, the robust stabilization problem is reduced to a convex feasibility problem involving linear matrix inequalities. The data necessary to assemble these inequalities can be readily obtained from a state-space model of the nominal plant.

1 Introduction

The subject of this paper is a robust stabilization problem for uncertain systems with real parametric uncertainty and dynamic uncertainty. The focus is on problems where the nominal design model is linear and time-invariant. The robust stabilization problem is to find a linear time-invariant controller that stabilizes all plants that result from perturbing the nominal model with bounded real parametric uncertainty and bounded unmodeled dynamics. We consider problems with output feedback. It is assumed that the open

*Supported in part by the National Science Foundation under YIA Grant no. ECS–93-58288.

loop transfer matrix of the nominal model, between the input and output channels for modeling uncertainty, has rank one for all frequencies. This is what we call *a system with rank-one uncertainty structure.* The results of this paper show that, under this assumption, a nonconservative solution to this class of problems can be obtained by solving a finite (probably large) number of linear matrix inequalities provided there is no restriction on the McMillan degree of the solution.

Rank-one robust stabilization problems, with mixed uncertainties, have been considered before by Rantzer and Megretski in a series of papers [6, 5, 7]. References [6, 7] show that the problems considered in this paper reduce to solving a convex feasibility problem in which the decision variables are given by two stable rational transfer matrices; i.e., an infinite-dimensional convex feasibility problem. This result is obtained with no *practical restriction* on the norms used to bound the parametric and dynamic uncertainty.

For the special case obtained when the parametric and dynamic uncertainty are correlated, and an l_2 norm bound is used to measure the size of the mixed uncertainty, reference [5] shows that the rank-one robust stabilization problem reduces to a convex linear matrix inequality problem whenever the decision variables are restricted to a finite dimensional subspace of the space of stable rational transfer matrices. This result is important because it opened the possibility of using recent commercial software [3] for computing solutions. This result has limited value due to the fact that the assumptions imposed on the uncertainty may not hold in all applications.

In this paper we show that finite-dimensional solutions to the optimization problems obtained in [6, 7] can be obtained in terms of *linear matrix inequality problems* with no practical limitations on the uncertainty description. More specifically, we show that for problems with uncorrelated parametric and dynamic uncertainty, a linear matrix inequality solution exists when an l_∞ or l_1 norm-bound on the parametric uncertainty is known.

The results presented here are only valid for systems with rank-one uncertainty structure; which is a restrictive assumption. However, we would like to point out that control problems with a single actuator do satisfy this assumption when the parametric uncertainty appears in the coefficients of the transfer functions between sensors and actuator in a linear fashion; see, for example, [7]. Of course, there are other situations where the rank-one assumption holds.

1.1 Notation

The notation used in this paper is fairly standard. The set of real (complex) numbers is denoted by R (C). For a complex matrix M, M' (M^*) denotes its transpose (conjugate transpose). If $M = M^*$ and $N = N^*$ are two complex matrices, the matrix inequality $M > N$ means that $M - N$ has strictly positive eigenvalues.

The notation

$$H(s) \stackrel{\text{SSR}}{=} \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right],$$

where A, B, C, and D are constant matrices, stands for state-space realization for the rational matrix function $H(s)$ and it means $H(s) = C(sI - A)^{-1}B + D$. A stable (all poles in the open left half-plane) rational matrix function H is strictly positive real (SPR) if $H(j\omega) + H^*(j\omega) > 0$ for all $\omega \in \mathrm{R} \cup \infty$.

2 Problem Statement

Consider the uncertain feedback system shown in Fig. 1, where the nominal plant transfer matrix is denoted by G and C is the transfer matrix of the controller we wish to find. The modeling uncertainties are denoted by δ and Δ, representing real parametric uncertainty (δ) and unmodeled dynamics (Δ), respectively.

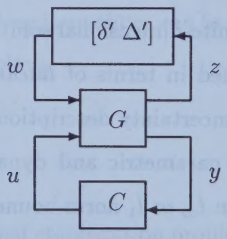


Fig. 1 Uncertain feedback system.

The key assumption made in this paper is that the the modeling uncertainty has rank one; this means that the dimension of the signal w in Fig. 1 is exactly one. In addition, the real parametric uncertainty δ belongs to the parameter set

$$\mathbf{\Delta}_{rs} = \{\delta \in \mathrm{R}^{p_0} \ : \ |\delta|_s \leq 1\}, \tag{1}$$

where $|\cdot|_s$ is the standard l_s norm in R^{p_0}. The unmodeled dynamics is described by complex constant perturbations in the set

$$\mathbf{\Delta}_c = \{\Delta \in \mathrm{C}^{p_1} \ : \ |\Delta|_2 \leq 1\}, \tag{2}$$

where $|\cdot|_2$ is the Euclidean norm (l_2) in C^{p_1}.

The problem studied in this paper is defined next.

Robust Stabilization Problem (RSP): *Find a finite dimensional real rational controller C such that the uncertain feedback system in Fig. 1 is internally stable for all $\delta \in \mathbf{\Delta}_{rs}$ ($s = 1$ or $s = \infty$) and for all $\Delta \in \mathbf{\Delta}_c$.*

3 Parametrization of solutions

Since $\delta = 0$ and $\Delta = 0$ are admissible values for the uncertainties, a solution C to RSP must also internally stabilize the nominal plant. Under this condition, it is well known [2] that the nominal closed loop transfer matrix T_{zw}, from w to z when δ and Δ are removed, obtained with the controller C may be written as

$$T_{z\omega} = \begin{bmatrix} T_{00} \\ T_{10} \end{bmatrix} + \begin{bmatrix} T_{01} \\ T_{11} \end{bmatrix} Q T_2 \tag{3}$$

where $T_{i0} \in RH_\infty^{p_i \times 1}$, $T_{i1} \in RH_\infty^{p_i \times m}$, $T_2 \in RH_\infty^{p_2 \times 1}$, and $Q \in RH_\infty^{m \times p_2}$. Moreover, when Q varies in $RH_\infty^{m \times p_2}$, formula (3) generates all closed loops that can be achieved with internally stabilizing controllers. This is true under the following mild assumption: the transfer matrix from u to y in G is strictly proper [2].

It becomes clear now that the RSP is equivalent to finding a rational matrix $Q \in RH_\infty^{m \times p_2}$ such that, for each $\delta \in \mathbf{\Delta}_{rs}$ and each $\Delta \in \mathbf{\Delta}_c$, the condition

$$1 - \delta'(T_{00} + T_{01} Q T_2) - \Delta'(T_{10} + T_{11} Q T_2) \neq 0 \tag{4}$$

holds at all points in the closed right half plane. To see this condition, simply compute the closed loop characteristic equation by breaking the loop at the signal w in Fig. 1 and use equation (3).

A condition that is useful for developing a numerical procedure for checking the existence of a rational matrix Q satisfying (4) is given next; the result is essentially from [7].

Theorem 3.1: *There exists* $Q \in RH_\infty^{m\times p_2}$ *such that, for each* $\delta \in \mathbf{\Delta}_{rs}$ *and each* $\Delta \in \mathbf{\Delta}_c$, *(4) holds in the closed right half plane if and only if there exist* $\alpha \in RH_\infty$ *and* $\beta \in RH_\infty^{m\times p_2}$ *such that*

$$-\operatorname{Re}\alpha(j\omega) + |\operatorname{Re}[T_{00}\alpha + T_{01}\beta T_2](j\omega)|_d + |[T_{10}\alpha + T_{11}\beta T_2](j\omega)|_2 < 0 \qquad (5)$$

for all $\omega \in \mathrm{R}\cup\{\infty\}$, *where* $|\cdot|_d$ *denotes the dual norm of* $|\cdot|_s$ *(i.e.,* $d^{-1} + s^{-1} = 1$*). Moreover, if (5) holds, a rational matrix* $Q \in RH_\infty^{m\times p_2}$ *satisfying (4) is given by* $Q = \alpha^{-1}\beta$.

Theorem 3.1 was given in [7] for the case of a scalar measurement (i.e., $p_2 = 1$). Here we observe that, for multiple measurements, the result still holds with no modification to the proofs. The function in the left-hand-side of (5) is a convex function of α and β. Thus, checking (5) amounts to solving an infinite-dimensional convex feasibility problem over the space of stable rational functions with real coefficients. This problem is only known to be solvable if the unknown functions α and β are approximated by functions from a finite dimensional subspace. This method is used in [7] for solving the resulting finite-dimensional convex programming problem with the ellipsoid algorithm.

For the case in which the uncertainties satisfy the Euclidean norm bound $\delta'\delta + \Delta^*\Delta \leq 1$, [5] gives a linear matrix inequality (LMI) characterization of the convex program. This LMI characterization leads to algorithms that are faster than the ellipsoid algorithm [1]. Our objective is to show that such LMI characterization exists for other practical uncertainty representations such as those in (1) and (2).

We first show that condition (5) holds if and only if certain rational matrices, linear in (α, β), are strictly positive real (SPR). To accomplish this, let $\mathbf{\Delta}_{rs}^v$ denote the set of extreme points of $\mathbf{\Delta}_{rs}$, i.e., $\mathbf{\Delta}_{rs}^v$ is the smallest finite set with the following property

$$\text{convex hull}(\mathbf{\Delta}_{rs}^v) = \mathbf{\Delta}_{rs}.$$

Theorem 4.1: $\alpha \in RH_\infty$ *and* $\beta \in RH_\infty^{m\times p_2}$ *satisfy (5) if and only if, for each* $\delta \in \mathbf{\Delta}_{rs}^v$, *the* $RH_\infty^{(1+p_1)\times(1+p_1)}$ *rational matrix*

$$\Phi(\alpha,\beta,\delta) = \begin{bmatrix} (1-\delta'T_{00})\alpha - \delta'T_{01}\beta T_2 & 0 \\ 2(T_{10}\alpha + T_{11}\beta T_2) & [(1-\delta'T_{00})\alpha - \delta'T_{01}\beta T_2]I \end{bmatrix} \qquad (6)$$

is SPR.

Proof: Notice first that, with $s = 1$ or $s = \infty$, we have

$$\begin{aligned} |\operatorname{Re}[T_{00}\alpha + T_{01}\beta T_2](j\omega)|_d &= \max_{\delta \in \mathbf{\Delta}_{rs}} \delta' \operatorname{Re}[T_{00}\alpha + T_{01}\beta T_2](j\omega) \\ &= \max_{\delta \in \mathbf{\Delta}_{rs}^v} \delta' \operatorname{Re}[T_{00}\alpha + T_{01}\beta T_2](j\omega). \end{aligned}$$

Hence, condition (5) in Theorem 3.1 is equivalent to the following condition: for each $\omega \in \mathrm{R} \cup \{\infty\}$ and each $\delta \in \mathbf{\Delta}_{rs}^v$, α and β satisfy

$$\operatorname{Re}\alpha(j\omega) - \delta' \operatorname{Re}[T_{00}\alpha + T_{01}\beta T_2](j\omega) > |[T_{10}\alpha + T_{11}\beta T_2](j\omega)|_2,$$

or, equivalently,

$$\operatorname{Re}\alpha(j\omega) - \delta' \operatorname{Re}[T_{00}\alpha + T_{01}\beta T_2](j\omega) > 0 \tag{7}$$

$$\operatorname{Re}\alpha(j\omega) - \delta' \operatorname{Re}[T_{00}\alpha + T_{01}\beta T_2](j\omega) > \frac{|[T_{10}\alpha + T_{11}\beta T_2](j\omega)|_2^2}{\operatorname{Re}\alpha(j\omega) - \delta' \operatorname{Re}[T_{00}\alpha + T_{01}\beta T_2](j\omega)}. \tag{8}$$

By using the Schur complement formula, at each fixed frequency ω, conditions (7) and (8) can be written as

$$\begin{bmatrix} \operatorname{Re}\alpha - \delta' \operatorname{Re}[T_{00}\alpha + T_{01}\beta T_2] & (T_{10}\alpha + T_{11}\beta T_2)^* \\ T_{10}\alpha + T_{11}\beta T_2 & \{\operatorname{Re}\alpha - \delta' \operatorname{Re}[T_{00}\alpha + T_{01}\beta T_2]\} I \end{bmatrix} (j\omega) > 0$$

which leads to the claim. □

We now show that when α and β are taken from a finite dimensional subspace of stable rational functions, the rational matrix $\Phi(\alpha, \beta, \delta)$ in (6) can be factored as the product of a constant real matrix, depending on the coefficients in the expansions of α and β, times a rational matrix that only depends on the data of the problem. To give such factorization, introduce the scalar RH_∞ function

$$f(s) := \frac{s - p}{s + p},$$

where p is a positive number, and define the rational vector

$$b'(s) := [1 \;\; f \;\; f^2 \;\; \cdots \;\; f^n],$$

where n is a fixed integer. Define $\alpha(s)$ and $\beta(s)$ to be of the form

$$\alpha(s) = x'b(s) \quad \text{and} \quad \beta(s) = \sum_{i=1}^{m} \sum_{j=1}^{p_2} y'_{ij} b(s) e_i e'_j \tag{9}$$

where x and y_{ij} are real vectors with $n+1$ components given by

$$x' = [x_0 \;\; x_1 \;\; \cdots \;\; x_n]$$

$$y'_{ij} = [y_{ij0} \;\; y_{ij1} \;\; \cdots \;\; y_{ijn}] \quad (i = 1 \cdots m,\; j = 1 \cdots p_2),$$

e_i denotes the standard i-th basis vector in R^m, and e_j denotes the standard j-th basis vector in R^{p_2}.

Lemma 4.2: *With $\alpha(s)$ and $\beta(s)$ given by (9), the rational matrix $\Phi(\alpha, \beta, \delta)$ in (6) can be written as*

$$\Phi(\alpha, \beta, \delta) = F(x, y_{11}, \ldots, y_{mp_2}) H_\delta(s) \tag{10}$$

where

$$F(x, y_{11}, \ldots, y_{mp_2}) = [I \otimes x' \;\; I \otimes y'_{11} \;\; \ldots \;\; I \otimes y'_{mp_2}] \tag{11}$$

$$H_\delta(s) = \begin{bmatrix} M^\delta(s) \\ N^\delta_{11}(s) \\ \vdots \\ N^\delta_{mp_2}(s) \end{bmatrix} \otimes b(s) \tag{12}$$

and

$$M^\delta(s) = \begin{bmatrix} (1 - \delta' T_{00}) & 0 \\ 2T_{10} & I \otimes (1 - \delta' T_{00}) \end{bmatrix}$$

$$N^\delta_{ij}(s) = \begin{bmatrix} -\delta' T_{01} e_i e'_j T_2 & 0 \\ 2T_{11} e_i e'_j T_2 & -I \otimes \delta' T_{01} e_i e'_j T_2 \end{bmatrix}.$$

Proof: Using elementary properties of Kronecker products, and the formulas in (9), we get

$$\begin{aligned} \Phi(\alpha, \beta, \delta) &= x' b(s) M^\delta(s) + \sum_{i=1}^{m} \sum_{j=1}^{p_2} y'_{ij} b(s) N^\delta_{ij}(s) \\ &= M^\delta(s) \otimes x' b(s) + \sum_{i=1}^{m} \sum_{j=1}^{p_2} N^\delta_{ij}(s) \otimes y'_{ij} b(s) \\ &= (I \otimes x')(M^\delta(s) \otimes b(s)) + \sum_{i=1}^{m} \sum_{j=1}^{p_2} (I \otimes y'_{ij})(N^\delta_{ij}(s) \otimes b(s)) \end{aligned}$$

$$= [I \otimes x' \;\; I \otimes y'_{11} \;\; \cdots \;\; I \otimes y'_{mp_2}] \begin{bmatrix} M^\delta(s) \otimes b(s) \\ N^\delta_{11}(s) \otimes b(s) \\ \vdots \\ N^\delta_{mp_2}(s) \otimes b(s) \end{bmatrix}$$

$$= [I \otimes x' \;\; I \otimes y'_{11} \;\; \cdots \;\; I \otimes y'_{mp_2}] \left(\begin{bmatrix} M^\delta(s) \\ N^\delta_{11}(s) \\ \vdots \\ N^\delta_{mp_2}(s) \end{bmatrix} \otimes b(s) \right).$$

This completes the proof. □

We now use Lemma 4.2 to give a state-space LMI that characterizes the positive real condition in Theorem 4.1. First, we need a minimal realization for the rational matrix $H_\delta(s)$ in (12), say

$$H_\delta(s) \stackrel{\text{SSR}}{=} \left[\begin{array}{c|c} A_\delta & B_\delta \\ \hline C_\delta & D_\delta \end{array}\right] \tag{13}$$

which must be computed for each extreme point $\delta \in \mathbf{\Delta}^v_{rs}$. A realization for $H_\delta(s)$ may be computed using the formulas in Lemma 4.2 and the realization formulas in [4] for the Kronecker product of transfer matrices. For the sake of brevity, further details are omitted.

From Lemma 4.2, and using (13), we get the following realization for $\Phi(\alpha, \beta, \delta)$:

$$\Phi(\alpha, \beta, \delta) \stackrel{\text{SSR}}{=} \left[\begin{array}{c|c} A_\delta & B_\delta \\ \hline F(x,y)C_\delta & F(x,y)D_\delta \end{array}\right]. \tag{14}$$

A straightforward application of the Positive Real Lemma [1] to the condition in Theorem 4.1, along with the realization in (14), gives the following result.

Theorem 4.3: *The rational functions α and β defined in (9) satisfy (5) if and only if the real vectors x, y_{11}, ..., y_{mp_2} satisfy the following property: for each $\delta \in \mathbf{\Delta}^v_{rs}$ there exists a real symmetric matrix P_δ such that*

$$\Psi_\delta(x, y_{11}, \ldots, y_{mp_2}, P_\delta) := \begin{bmatrix} A'_\delta P_\delta + P_\delta A_\delta & C'_\delta F'(x, y_{11}, \ldots, y_{mp_2}) - P_\delta B_\delta \\ F(x, y_{11}, \ldots, y_{mp_2})C_\delta - B'_\delta P_\delta & -F(x, y_{11}, \ldots, y_{mp_2})D_\delta - D'_\delta F'(x, y) \end{bmatrix} < 0. \tag{15}$$

We conclude this section with a summary of the key steps required to check the existence of solutions to the Robust Stabilization Problem defined in section 2.

Procedure

1. Compute a stabilizing controller C_0 for the nominal plant G in Fig. 1 and determine the corresponding rational matrices T_{ij} in equation (3) using standard state-space computations [2]. Select an integer n—the "order of the expansions" in (9).

2. Use the formula (12) to obtain, for each $\delta \in \mathbf{\Delta}_{rs}^{v}$, a minimal realization for the rational matrix $H_\delta(s)$.

3. Assemble the system of linear matrix inequalities

$$\Psi_{\delta_1}(x, y_{11}, \ldots, y_{mp_2},\ P_{\delta_1}) < 0, P_{\delta_1} = P'_{\delta_1}$$
$$\Psi_{\delta_2}(x, y_{11}, \ldots, y_{mp_2},\ P_{\delta_2}) < 0, P_{\delta_2} = P'_{\delta_2}$$
$$\ldots$$
$$\Psi_{\delta_L}(x, y_{11}, \ldots, y_{mp_2},\ P_{\delta_L}) < 0,\ P_{\delta_L} = P'_{\delta_L}$$

   where $\Psi_\delta(\cdot)$ is defined in (15), L is the cardinality of $\mathbf{\Delta}_{rs}^{v}$, and $\delta_1, \ldots, \delta_L$ denote the elements of $\mathbf{\Delta}_{rs}^{v}$. Compute a solution to this system of inequalities.

4. Using a solution x, $y_{11}, \ldots, y_{mp_2}$ to the inequalities in step 3, compute $\alpha(s)$ and $\beta(s)$ from equation (9). Compute Q using the formula $Q = \alpha^{-1}\beta$.

5. Use the solution Q from step 4, and standard formulas for the stabilizing controllers [2], to find a solution C_{sol} for the Robust Stabilization Problem.

It should be noted that the system of matrix inequalities in step 3 above is a standard system of linear matrix inequalities; thus, its feasibility may be easily tested using standard software [3]. Note also that the system of inequalities may have no solution. In this case, one should increase the integer n and repeat the procedure. If for high n no solution exists, then it is advisable to attempt the problem with smaller (less than 1) norm bounds for the uncertainty, see equations (1) and (2). A solution always exist when there is no uncertainty.

4 Conclusion

We have considered a robust stabilization problem with both real parametric and dynamic uncertainties. Under a special rank-one assumption we have shown that the existence of solutions to this class of problems can be characterized with a finite number of LMIs. The main contribution of this paper is to show that the LMI characterization of the robust stabilization problem is possible under assumptions, on the uncertainty description, which are less restrictive than the assumptions used before.

References

[1] S. Boyd, L. E. Ghaoui, E. Feron and V. Balakrishnan, " Linear Matrix Inequality in System and Control Theory," SIAM, 1994.

[2] B. Francis. "A course in $\mathcal{H}_\infty$ control theory," volume 88 of *Lecture Notes in Control and Information Sciences.* Springer-Verlag, New York, 1987.

[3] P. Gahinet, Nemirovskii, A., Laub, A. J., and Chilali, M., *LMI Control Toolbox*, The MathWorks Inc., 1994.

[4] P.P. Khargonekar and M.A. Rotea, "Multiple objective optimal control of linear systems: The quadratic norm case," *IEEE Trans. Automat. Control,* vol. AC-36, no. 1, 1991, pp. 14-24.

[5] A. Rantzer, "Linear Matrix Inequalities for Rank One Robust Synthesis," *Proc. of the IEEE CDC*, 1993, pp. 2590-2591.

[6] A. Rantzer and A. Megretski, "A Convex Parameterization of Robustly Stabilization Controllers, " *Robustness of Dynamic Systems with Parameter Uncertainties*, M. Mansour, S. Balemi and W. Truol Eds., Birkhauser Verlag Basel, 1992, pp. 271-279.

[7] A. Rantzer and A. Megretski, "A Convex Parameterization of Robustly Stabilizing Controllers, " *IEEE Trans. on Auto. Contr.*, 1994, pp. 1802-1808.

ROBUST STABILITY OF UNCERTAIN FUNCTIONAL DIFFERENTIAL EQUATIONS

A. S. C. Sinha
Purdue University at Indianapolis
Department of Electrical Engineering
SL 160, Indianapolis, IN 46202

ABSTRACT

In this paper, we investigate the control of uncertain functional differential equations with variable coefficients and variable delays under the assumptions of continuity and boundedness of the coefficients and continuous differentiability of the delays. The results are derived by variable structure control method. The stability of uncertain system in the sliding mode is discussed. Our main result shows that the variable structure control can be used to stabilize uncertain nonlinear delay differential systems. Sufficient conditions are also derived when the solutions of nonlinear uncertain delay differential systems are dominated by the solutions of comparison systems.

INTRODUCTION

Variable structure control of dynamic systems described by ordinary differential equations have undergone a wealth of theoretical developments in the recent years. Systematic study of the variable structure control can be found in [1-5] among many others. Some results for the control of dynamic systems described by time-delay differential equations have also appeared in the last few years in [6-7].

The purpose of this paper is to study the variable structure control of an uncertain functional differential equations

$$\dot{x}(t) = \sum_{i=1}^{N} A_i(t)x(t-\tau_i(t)) + f(t,x(t-\tau_i(t)) + B(t)u(t), \qquad (1)$$

where x(t) is an n-dimensional vector of the system state; $A_i(t)$ (i = 1,..., N), are given n × n bounded variable matrices. The elements of B(t) are n × m real continuous functions specified for $t \geq t_0$. . u(t) is an m-dimensional vector of control input. f is an n-dimensional vector of unknown function of delayed states. The scalar functions $\tau_i(t)$ are called time delays and are specified for all i = 1,..., n, such that (i) $\tau_i(t)$ is a function of bounded variation and is right continuous on each bounded interval $[0, \tau] \subset [0, \infty)$, (ii) $\tau_i(t) \geq 0$ for all $t \in [0, \infty)$, and (iii) $t - \tau_i(t)$ is monotonic non-decreasing. A great deal of attention has been devoted to the delay differential systems of the form (1). The hypotheses and results are motivated by the study of systems

$$\dot{x}(t) = Ax(t-\tau(t)) \qquad (2)$$

considered in [8] with time-varying delays (and no inputs and no uncertainty). Parks and Miller [9] have considered delay-differential equations of the form

$$\dot{x}(t) = \sum_{j=1}^{n} a_{ij} x_j(t - h_{ij}), \quad (i = 1,2,\ldots,n) \tag{3}$$

which has been thoroughly investigated as a model for pulse code modulation systems that are digital telephone networks involving an oscillator at each telephone exchange and serves as a clock [5]. Cooke [10] has a derivation of such model for infection.

DEFINITION 1. The unique solution of the initial value problem (1) is denoted by $x(t_0, \varphi)(t)$ or $x_t(t_0,\varphi)$ depending upon whether we view solution in E or $C \equiv C([-\tau, 0], E]$. We recall that for a function $x \in C([-\bar{\tau}, \infty), E)$, x_t denotes the elements of C given by $x_t(s) = x(t + s)$, $-\bar{\tau} \le s \le 0$.

DEFINITION 2. A domain D contained in the switching surface S is a sliding mode domain if for each $\varepsilon > 0$ there exist a $\delta > 0$ such that any solution $x(t_0, \varphi)(t)$ starting in the n-dimensional δ-neighborhood of D may leave the n-dimensional ε-neighborhood of D only through the n-dimensional ε-neighborhood of the boundary of D.

Note that a trajectory starting in D can leave D only through the boundary in the switching surface S.

ROBUST STABILITY OF UNCERTAIN TIME-DELAY SYSTEMS

Let us assume for simplicity that the uncertain nonlinear function $f(t, x(t - \tau_i(t))$ is bounded and satisfy the linearity condition

$$f(t, x(t - \tau_i(t)) = \sum_{i=1}^{N} \Delta A_i(t) x(t - \tau_i(t)) \tag{4}$$

If we substitute equation (4) in equation (1), then we have

$$\dot{x}(t) = \sum_{i=1}^{N} (A_i(t) + \Delta A_i(t))\, x(t - \tau_i(t)) + B(t)u(t) \tag{5}$$

Equation (5) is rewritten as

$$\dot{x}(t) = \sum_{i=1}^{N} \hat{A}_i(t)\, x(t - \tau_i(t)) + B(t)u(t) \tag{6}$$

where $\hat{A}_i(t) \equiv A_i(t) + \Delta A_i(t)$. Assume that there exists a matrix $H_i(t) \in E^{1\times n}$ such that $B(t) \otimes H_i(t) = \Delta A_i(t)$ for each $i = 1,\ldots,N$, then we select the switching function $S(t)$ as

$$S(t) = Qx(t) + \int_0^t QA_0(\alpha)x(\alpha)d\alpha - \sum_{i=1}^{N} \int_0^t QA_i(\alpha)x(t - \tau_i(\alpha))d\alpha \tag{7}$$

where $S(t) \in E^{m\times 1}, Q \in E^{m\times n}$. $A_0(t)$ is suitably chosen to stabilize system after it enters in sliding mode. $u_{eq}(t)$ is obtained by differentiating $S(t)$ with respect to time and letting $\dot{S}(t) = 0$. We obtain

$$u_{eq}(t) = -(B(t))^{-1} A_0(t)x(t) - \sum_{i=1}^{N} H_i(t)\, x(t - \tau_i(t)) \tag{8}$$

If we select $u(t)$ as follows

$$u(t) = u_{eq}(t) - \mu\, (Q \otimes B)^{-1}\, \mathrm{sgn}(S(t)); \ \mu > 0 \tag{9}$$

where Q⊗B(t) is nonsingular for some Q. Then the switching function *S(t)* defined in equation (7) satisfy the condition

$$\begin{aligned} S^T(t) \otimes \dot{S}(t) &= -\mu[s_1, s_2, \cdots, s_m]\mathrm{sgn}(S(t)) \\ &= -\mu \sum_{i=1}^{m}(|s_i|) \le 0; \quad \mu > 0 \end{aligned} \tag{10}$$

From (10), we know that the sliding mode condition is always satisfied, that means the trajectories will reach the switching surface S(t) = 0, whenever we add the control law (9) to the system (1). After the system enters in the sliding mode, the equivalent system dynamic equation is obtained by substituting $u_{eq}(t)$ from the equation (8) in (6) yields, after some computation

$$\dot{x}(t) = -A_0(t)x(t) + \sum_{i=1}^{N}[A_i(t)\, x(t - \tau_i(t))] \tag{11}$$

If the equivalent sliding mode dynamic equation (11) is asymptotically stable, satisfying the sliding mode conditions $S^T(t) \otimes \dot{S}(t) \le 0$, then it follows that the solution of (11) is asymptotically stable, driven by the sliding mode controller (8). For the stability of the trivial solution of (11), We use the functional V: $\Re \times C \rightarrow C$ of the form

$$V(t, \varphi) = \left[\sum_{i=1}^{n}\varphi_i^2(0)\right] + \sum_{i=1}^{n}\sum_{k=1}^{N}\int_{-\tau_k(t)}^{0} C_i(t+\theta)\varphi_i^2(\theta)\, d\theta, \tag{12}$$

where $C_i(t) : \Re \rightarrow \Re$, are continuous n × n matrix to be chosen below. Then a calculation yields $\dot{V}(t,\varphi)$, viewed as a quadratic form in φ(0) and φ(− τ_i(t)), i= 1,…,N

$$\begin{aligned} -\dot{V}(t,\varphi) = &\sum_{i=1}^{n}\sum_{j=1}^{n}\left[2A_{0ij}(t) - \sum_{k=1}^{N}C_{kij}\right]\varphi_i(0)\varphi_j(0) \\ &+ \sum_{i=1}^{n}\sum_{k=1}^{N}(1 - \dot{\tau}(t))C_i(t+\theta)\varphi_i^2(-\tau_k(t)) \\ &2\sum_{i=1}^{n}\sum_{j=1}^{n}\sum_{k=1}^{N}A_{kij}(t)\varphi_i(0)\varphi_j(-\tau_k(t)) \end{aligned} \tag{13}$$

It has the following associated symmetric matrix

$$M(t) = \begin{bmatrix} A_0^*(t) - \sum_{i=1}^{N}C_i(t) & -\frac{1}{2}A_1^*(t) & -\frac{1}{2}A_2^*(t) & \cdots & -\frac{1}{2}A_N^*(t) \\ -\frac{1}{2}A_1^*(t) & D_1(t) & 0 & \cdots & 0 \\ -\frac{1}{2}A_2^*(t) & 0 & D_2(t) & 0 & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -\frac{1}{2}A_N^*(t) & 0 & 0 & \cdots & D_N(t) \end{bmatrix} \tag{14}$$

where let

$$\begin{aligned} &A_0^*(t) \equiv (A_0(t) + A_0^T(t)),\ A_i^*(t) = (A_i(t) + A_i^T(t)); \quad i = 1,\ldots,N \\ &D_i(t) \equiv (1 - \dot{\tau}_i(t))C_i(t - \tau_i(t)), \quad i = 1,\ldots,N \end{aligned} \tag{15}$$

and I_n is the unit n × n matrix. We now choose

$$C_i(t) = \frac{\|A_i(t + \tau_i(t))\|}{\alpha_i}, \; i = 1,2,\ldots,N \tag{16}$$

The quadratic form associated with the matrix M(t) will be non-negative if the Sylvester conditions

$$\Delta_1(t) \geq 0; \quad \Delta_2(t) \geq 0; \ldots; \Delta_N(t) \geq 0 \tag{17}$$

are satisfied. Δ_i are defined as the principal minors of the matrix M(t). It is well known that the quadratic for is positive definite if $\Delta_i > \alpha_0$; (i = 1, …, n); where $\alpha_0 > 0$ is a constant. Let us define

$$\begin{aligned}
\Delta_1(t) &= (A_0(t) + A_0^T(t)) - \sum_{i=1}^{N} \frac{1}{\alpha_i} \|A_i(t+\tau_i(t))\|, \\
\Delta_2(t) &= (1-\dot{\tau}_1(t) I_n \left[\frac{\|A_1(t)\|}{\alpha_1} \right] \left[A_0(t) + A_0^T(t)) - \alpha_1 \|A_1(t)\| - \sum_{i=1}^{N} \frac{1}{\alpha_i} \|A_i(t+\tau_i(t))\| \right], \\
&\cdots \\
\Delta_N(t) &= \prod_{i=1}^{N} (1-\dot{\tau}_i(t)) I_n \left[\frac{\|A_i(t)\|}{\alpha_i} \right] \\
&\otimes \left[(A_0(t) + A_0^T(t)) - \sum_{i=1}^{N} \alpha_i \|A_i(t)\| I_n - \sum_{i=1}^{N} \frac{1}{\alpha_i} \|A_i(t+\tau_i(t))\| \right]
\end{aligned} \tag{18}$$

for some constants $\alpha_0 > 0$, and $\alpha_i > 0$, for i = 1, …, N. If $\dot{\tau}_i(t) < 1$, for all i and t, then for the principal minors to be positive, the matrix $A_0(t)$ must be chosen such that

$$\left[(A_0(t) + A_0^T(t)) - \sum_{i=1}^{N} \alpha_i \|A_i(t)\| I_n - \sum_{i=1}^{N} \frac{1}{\alpha_i} \|A_i(t+\tau_i(t))\| \right] \geq \alpha_0 I_n > 0 \tag{19}$$

where $\alpha_0 > 0$ and I_n is the identity matrix of size n × n. Using the standard arguments, we conclude that if (19) hold, then there exists $\alpha_0 > 0$ such that

$$\dot{V}(t,\varphi) \leq -\alpha_0 \|\varphi^2(0)\| \tag{20}$$

for all $\varphi \in C$. Moreover, $V(t, \varphi) \geq \| \varphi(0) \|^2$ and

$$V(t,\varphi) \leq \|\varphi(0)\|^2 + \|\varphi\|_\infty^2 \sum_{i=1}^{N} \int_t^{t+\bar{\tau}} \|A_i(s)\|^2 \, ds, \quad \sup_t[\tau_i(t)] = \bar{\tau}, \quad \text{for all i} \tag{21}$$

If

$$\sum_{i=1}^{N} \int_t^{t+\bar{\tau}} \|A_i(s)\|^2 \, ds \leq \alpha < \infty \tag{22}$$

then, we have

$$\begin{aligned}
\|\varphi(0)\|^2 \leq V(t,\varphi) &\leq \|\varphi(0)\|^2 + \|\varphi\|_\infty^2 \sum_{i=1}^{N} \int_t^{t+\bar{\tau}} \|A_i(s)\|^2 \, ds \\
&\leq \|\varphi\|_\infty^2 (1+\alpha)
\end{aligned} \tag{23}$$

which implies that this functional satisfies the conditions of the Krasovskii stability theorem.

THEOREM 1. Consider the system (11) and assume that there exists matrix $H_i(t)$ such that condition $B \otimes H_i = \Delta A_i$ is satisfied for each i, and if $A_0(t)$ in (7) is chosen such the conditions of inequality

$$\text{(i)} \quad \left[(A_0(t) + A_0^T(t)) - \sum_{i=1}^{N} \alpha_i \|A_i(t)\| I_n - \sum_{i=1}^{N} \frac{1}{\alpha_i} \|A_i(t+\tau_i(t))\| \right] \geq \alpha I_n > 0$$

and

$$\text{(ii)} \quad \sum_{i=1}^{N} \int_t^{t+T} \|A_i(s)\|^2 \, ds \leq \alpha < \infty$$

hold. Then the solution of the system (1) driven by controller u(t) (9) is asymptotically stable.

COMPARISON THEOREM

Let the coefficients of the matrices $A_0(t)$, and $A_i(t)$ are defined by the sliding mode equation (11). Let the coefficient matrix $A_0(t)$ in equation (8), is chosen as a diagonal matrix, Let us assume that $A_i(t)$ is non-negative. Then the sliding mode equation (11) is written as

$$\dot{x}(t) = -A_0(t)x(t) + \sum_{i=1}^{N} A_i(t)z(t - \tau_i(t)) \tag{24}$$

The elements of each $n \times n$ matrix $A_i(t)$ are non-negative and are continuous functions of real variable t on $0 \le t < \infty$, and the matrix $A_0(t)$ is chosen as a diagonal matrix. Consider a delay-differential comparison system

$$\dot{z}(t) = -A_0(t)z(t) + \sum_{i=1}^{N} A_i(t)z(t - \bar{\tau}) \tag{25}$$

Suppose that the continuous lags $\tau_i(t)$ are related to those of (11) by

$$t > t - \tau_i(t)) \ge t - \bar{\tau} \tag{26}$$

In Theorem 2, following [5], we assume that the frequencies' $x_1(t)$, ..., $x_n(t)$ of n oscillators satisfy equation (24), where the coefficients $a_{ii} > 0$ in diagonal matrix $A_0(t)$. The off diagonal elements of matrices $A_i(t)$ (i = 1,...,N) are either zero or equal to $1 > 0$ and each non-zero a_{ij} corresponds to a direct link joining the j-th telephone to the i-th. The time lag $\tau_{ij}(t)$ is proportional to the length of landline joining these exchanges but it can vary with temperature changes. $f(\cdot)$ is an uncertainty function considered as before. Equation (24) is a model for pulse code modulation systems that are digital telephone networks involving an oscillator at each telephone exchange that serves as a clock and has been considered by many with no inputs and no uncertainty.

For real matrices $A = (a_{ij})$, $B = (b_{ij})$, define $A \ge B$ mean $a_{ij} \ge b_{ij}$ for all pairs (i, j). Define $|A|$ to be the matrix $(|a_{ij}|)$. These definitions also apply for column or row vectors. Define a diagonal matrix C(t) by

$$C(t) = \int_{-\bar{\tau}}^{t} A_0(s)ds \tag{27}$$

Next, we state the following theorem:

THEOREM 2. Assume that there exists matrix $H_i(t)$ such that condition $B \otimes H_i = \Delta A_i$ is satisfied, and if $A_0(t)$ in (7) is chosen such that the coefficients of the systems (24) and (25) are related for each i

$$\text{(i)}\ \|A_i(t)\| \left[\exp \int_{t-\bar{\tau}}^{t-\tau_i(t)} A_0(s)ds \right] \le A_i(t),$$

And if,

$$\text{(ii)}\ \|z(t)\| < \left[\exp \int_{t}^{\theta} A_0(s)ds \right] x(\theta),$$

on the initial set $-\bar{\tau} \le t \le \theta \le 0$, then $\|z(t)\| < x(t)$ for all $t \ge 0$. In addition if the hypotheses of theorem 1 hold, then the solution of the system (24) driven by controller u(t) (9) is asymptotically stable provided there exists a locally integrable function $\mu(t)$ such that the eigenvalues of the symmetric matrix

(iii) $\frac{1}{2}[A_0^T(t)P(t) + P(t)A_0(t) + \dot{P}(t) + \upsilon I_n] \leq \mu(t)$

and satisfying

(iv) $\lim_{t\to\infty} \int_{t_0}^{t} \mu(s)ds = -\infty$

Proof. Since the condition (ii) on the initial set $-\bar{\tau} \leq t \leq \theta \leq 0$ satisfy the inequality

$$e^{C(t)}\|z(t)\| < e^{C(\theta)}x(\theta) \tag{28}$$

we claim that $\|z(t)\| < x(t)$ for $0 \leq t \leq T$. If this is false then the continuity of the solution gives a number t^* in $0 < t^* \leq T$ such that

$$\|z(t)\| < x(t) \text{ for } 0 < t < t^*, \text{ but } \|z(t^*)\| \not< x(t^*) \tag{29}$$

which from (28) that $x(t) > 0$ for $-\bar{\tau} \leq t \leq t^*$.Now consider the derivative

$$\frac{d}{dt}\left[e^{C(t)}x(t)\right] = \sum_{i=1}^{N} e^{C(t)}A_i(t)x(t-\tau_i(t)) \tag{30}$$

which is non-negative in this range. Then $e^{C(t)}x(t)$ is monotonic increasing since its derivative since $x(t - \tau_i(t)) > 0$ in this range and each element of $A_i(t)$ is non-negative. Therefore

$$e^{C(\theta)}x(\theta) \geq e^{C(t)}x(t) > e^{C(t)}z(t), \quad (t^* > \theta \geq t \geq 0) \tag{31}$$

which implies that (29) holds for $-\bar{\tau} \leq t \leq \theta < t^*$. Now, consider equation (25), its solution can be written as

$$e^{C(t^*)}z(t^*) = e^{C(0)}z(0) + \sum_{i=1}^{N}\int_0^{t^*} e^{C(t)}A_i(t)z(t-\bar{\tau})dt \tag{32}$$

Estimating (32) with respect to norm, we obtain

$$e^{C(t^*)}\|z(t^*)\| \leq e^{C(0)}\|z(0)\| + \sum_{i=1}^{N}\int_0^{t^*} e^{C(t)}\|A_i(t)\|z(t-\bar{\tau})\|dt \tag{33}$$

Now apply condition (ii) 0f theorem 2, (29), and (30) to get

$$\begin{aligned} e^{C(t^*)}\|z(t^*)\| &< e^{C(0)}x(0) + \sum_{i=1}^{N}\int_0^{t^*} e^{C(t)}\|A_i(t)\|\, x(t-\tau_i(t))dt \\ &\leq e^{C(0)}x(0) + \int_0^{t^*}\frac{d}{dt}\left[e^{C(t)}x(t)\right]dt \\ &\leq e^{C(t^*)}x(t^*) \end{aligned} \tag{34}$$

Hence $\|z(t^*)\| < x(t^*)$ for $0 \leq t < t^*$ which contradicts that $\|z(t^*)\| \not< x(t^*)$.This contradiction establishes the assertion that the solution of (24) is bounded by the solution of (25). If the solutions of (25) is asymptotically stable then the solution of (24) is also asymptotically stable. Consider the equation (25), where we assume that there exists a Cauchy matrix $C(t, \tau)$ for the auxiliary system $\dot{x}(t) = A_0(t)x(t)$ such that

$$\|C(t,\tau)\| \le Ke^{-\alpha(t-\tau)}, \quad K>0, \alpha>0 \tag{35}$$

for all t, $\tau \ge 0$. Malkin has proved that if the auxiliary system is uniformly asymptotically stable and if $\tilde{A}_0(t)$ is uniformly bounded on [0, ∞), then the following quadratic form

$$V(t, z(t)) = z^T(t)\, P(t)\, z(t) = \int_t^{\infty} \|C(t,\tau)z(t)\|^2 d\tau, \tag{36}$$

is a Liapunov function such that

$$\lambda^2 \| z(t)\|^2 \le V(t, z(t)) \le \Lambda^2 \| z(t)\|^2, \quad \Lambda > \lambda > 0 \tag{37}$$

where P(t) is positive definite symmetric matrix. A straight forward calculation with respect (36) yields

$$\begin{aligned} \dot{V}(t,z_t) = &[z^T(t)[A_0^T(t)P(t) + P(t)A_0(t) + \dot{P}(t)]z(t) \\ &+[\sum_{i-1}^{N} z^T(t-\tau_i)A_i^T(t)P(t)z(t) + \sum_{i=1}^{N} z^T(t)P(t)A_i(t)z(t-\tau_i)] \end{aligned} \tag{38}$$

Suppose there exist a constant $\eta > 1$ such that Razumikhin condition [11].

$$\eta^2 [z^T(t)\, P(t)\, z(t)] > [z^T(s)\, P(s)\, z(s)] \quad \text{for } 0 \le s \le t, \ \eta > 1 \tag{39}$$

hold. This implies that

$$\eta^2 [z^T(t)\, P(t)\, z(t)] > \lambda^2 \| z_t \|^2 \tag{40}$$

Let $\| A_i(t)\| \le a_i$, $a = \sum_{i=1}^{N} a_i$, $t_0 \le t < \infty$ then using (40), we obtain

$$\begin{aligned} &[\sum_{i=1}^{N} z^T(t-\tau_i)A_i^T(t)P(t)z(t) + \sum_{i=1}^{N} z^T(t)P(t)A_i(t)z(t-\tau_i)] \\ &\le \sum_{i=1}^{N} 2\|A_i(t)\|z(t)z(t-\tau_i) \\ &\le 2a\|z_t\|\,\|z(t)\| \\ &\le 2a\eta\left[\frac{(z^T(t)P(t)z(t)}{\lambda^2}\right]^{1/2}\|z(t)\| \\ &\le 2a\eta\|z(t)\|^2\left(\frac{\Lambda}{\lambda}\right) \end{aligned} \tag{41}$$

The equation (38) can be written as

$$\dot{V}(t,z_t) \le z^T(t)[A_0^T(t)P(t) + P(t)A_0(t) + \dot{P}(t) + \upsilon I_n]z(t) \tag{42}$$

where $\mu \equiv \frac{2\hat{a}\eta\Lambda}{\lambda}$, I_n is n × n unity matrix. Since P(t) is positive definite and symmetric, there exists $\lambda > 0$, such that $V(t, z(t)) \ge \lambda^2 \|z(t)\|^2$. Estimating $\dot{V}(t,z(t))$ by (42), we obtain for the following inequality

$$\dot{V}(t,z(t)) \le 2\,\mu(t)\, V(t, z(t)) \tag{43}$$

from which the desired conclusion follows by a direct integration. This proves the asymptotic stability of the sliding mode dynamics given by equation (24).

APPLICATION

As an application of the previous theory, let us consider a special class of equations with delays; namely the system of equations

$$\dot{x}(t) = \sum_{k=1}^{N} A_k x(t-\tau_k(t)) + f(t, x(t-\tau_k) + B\,u(t) \tag{44}$$

where each A_k is a constant n × n matrix, $0 \le \tau_k(t) \le \bar{\tau}$, and f is an n-dimensional vector of unknown function of delayed states. The switching function is defined from equation (7) as where A_0 is chosen as a real constant n × n matrix, the eigenvalues of which all have negative real parts. $\|A_i(\theta)\| \le \eta(\theta)$ for $\theta \in -\bar{\tau} \le t \le 0$, $\eta \in L^1[0, \infty)$. The matrix Q is choose such that $Q \otimes B = 1$. Let us assume for simplicity that the uncertain nonlinear function f(t, x(t $-\tau_i(t)$) is bounded and satisfy the linearity condition

$$f(t, x(t-\tau_i(t)) = \sum_{i=1}^{N} \Delta A_i(t) x(t-\tau_i(t)) \tag{46}$$

If we substitute equation (46) in equation (44), then we have

$$\dot{x}(t) = \sum_{i=1}^{N} (A_i(t) + \Delta A_i(t))\, x(t-\tau_i(t)) + B(t)u(t) \tag{47}$$

$u_{eq}(t)$ is obtained by differentiating the switching function S(t) with respect to time and letting $\dot{S}(t) = 0$ as in equation (7)

$$S(t) = Qx(t) + \int_0^t QA_0(\alpha)x(\alpha)d\alpha - \sum_{i=1}^{N} \int_0^t QA_i(\alpha)x(t-\tau_i(\alpha))d\alpha \tag{48}$$

We obtain

$$u_{eq}(t) = -(B(t))^{-1} A_0 x(t) - \sum_{i=1}^{N} H_i(t)\, x(t-\tau_i(t)) \tag{49}$$

Then u(t) is selected as

$$u(t) = u_{eq}(t) - \mu\,(Q \otimes B)^{-1}\, \mathrm{sgn}(S(t));\ \mu > 0 \tag{50}$$

As shown earlier the sliding regime $(S(t) \otimes \dot{S}(t) \le 0)$ is always satisfied for the switching function (48), which means hitting the switching surface ($S = 0$) will always occur when we add control law u(t). After the system enters in the sliding mode, the equivalent system dynamic equation is obtained by substituting $u_{eq}(t)$ from the equation (49) in (47) yields, after some computation

$$\dot{x}(t) = -A_0 x(t) + \sum_{i=1}^{N} [A_i(t)\, x(t-\tau_i(t))] \tag{51}$$

Let us choose $V(x(t)) = x^T(t)\, P\, x(t)$, then V(t,x(t) satisfy the inequality

$$\lambda \|x(t)\|^2 \le V(x(t)) \le \Lambda\, \|x(t)\|^2 \tag{52}$$

where λ and Λ are the least and the greatest eigenvalues of P. There exist a matrix P such that $[A_0^T P + P A_0] = -I$. Define $\|x_t\| = \sup_{0 \le s \le t} \|x(s)\|$. By Razumikhin's criterion, there exists a constant $\upsilon > 1$ such that

$$\upsilon^2 [x^T(t) P x(t)] > [x^T(s) P x(s)] \quad \text{for} \quad 0 \le s \le t \tag{53}$$

This implies that $\upsilon^2 [x^T(t) P x(t)] > \lambda^2 \|x_t\|^2$ .A strightforward calculation yields

$$\dot{V}(t,x_t) = [x^T(t)[A_0^T P + PA_0]z(t)$$

$$+[\sum_{i-1}^{N} x^T(t-\tau_i)A_i^T(t)Px(t) + \sum_{i=1}^{N} x^T(t)PA_i(t)x(t-\tau_i)]$$

$$= -\|x(t)\|^2 + [\sum_{i-1}^{N} x^T(t-\tau_i)A_i^T(t)Px(t) + \sum_{i=1}^{N} x^T(t)PA_i(t)x(t-\tau_i)] \qquad (54)$$

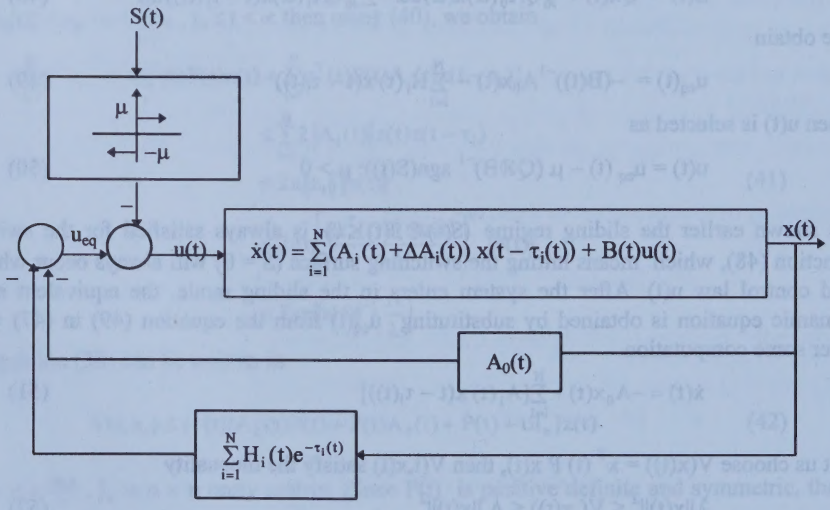
$$-\|x(t)\|^2 + \left[2\sum_{i=1}^{N}\|A_i(t)\|\|x(t)\|\|x_t\|\right]$$

$$\leq -\|x(t)\|^2 + \left[2\sum_{i=1}^{N}\|A_i(t)\|\left[\frac{\Lambda}{\lambda}\right]\right]\|x(t)\|^2$$

From the condition (ii) of the theorem 1, we have

$$\sum_{i=1}^{N}\int_t^{t+T}\|A_i(s)\|^2 ds \leq \alpha < \infty$$

then $\upsilon > 1$ can be found such that $2\upsilon\alpha\left[\frac{\Lambda}{\lambda}\right] < 1$. We have shown that the solution of the system (44) driven by controller u(t) (50) is asymptotically stable. A block diagram representation of the closed loop system for equations in sliding mode is given in figure below.



Figure

REFERENCES

1. Utkin V. I., Sliding modes in control optimization, *Springer-Verlag*, Berlin, New York, (1992).

2. Fliess M., Generalized controller canonical forms for linear and nonlinear dynamics, *IEEE Trans. on Automatic Control*. vol. **35**, 994-1001 (1990).

3. Nijmeijer H., & A. Van der Schaft, Nonlinear dynamical control systems, *Springer-Verlag*, Berlin-New York, (1990).

4. Sira-Ramirez H., Asymptotic output stabilization for nonlinear systems via dynamical variable structure control, *Dynamics and Control,* vol. **1**. 45-58 (1991).

5. Slotine J.J. E., Sliding controller design for nonlinear systems, *International J. of Control*, vol. **40**, 421-434 (1984).

6. Sinha A. S. C., El-Sharkawy M. A. & Rizkalla M. E., Moving Control for Time-Varying Linearizable State-Space Systems, *Proc. of the ISCA International Conference*, 295-297 (1995).

7. Drakunov S. V. & Utkin V. I., Sliding mode control in dynamic systems, *Int. J. of Control*, vol. **55**, 1029-1037 (1991).

8. Györi I. Necessary and sufficient stability conditions in an asymptotically ordinary delay differential equation, *Differential and Integral Equations,* vol. **6**, 225-239 (1993).

9. Parks P. C. and Miller M. R. Stability of p. c. m. characteristic equations, *Electronic Letters,* 3, 193-194, (1967).

10. Cooke K. L. Functional differential systems: Some models and perturbation problems, *International Symposium on Differential Dynamical Systems,* Academic Press, New York, 167-183, 1965.

11. Razumikhin, B. S. Application of Liapunov method to problem in the stability of systems with a delay, *Automat. I. Telemeh.* 21, 740-749, 19960.

Robust Performance Bounds based on Lyapunov Functions for Uncertain Systems

V. Balakrishnan

School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907-1285, Internet: ragu@ecn.purdue.edu

Abstract

The robust stability analysis of uncertain systems, with various assumptions on the nature of the uncertainties (sector-bounded nonlinear, linear time-invariant, parametric, etc.), as well as their structure (diagonal, block-diagonal, etc.), can be performed in a unified manner using multiplier theory and LMI-based convex optimization. The multipliers used in the stability analysis can be shown to yield a convex parametrization of a subset of Lyapunov functions that provide a certificate of robust stability. We show how these Lyapunov functions can in turn be used derive bounds on various robust performance measures of the system. We illustrate our approach with three specific robust performance analysis problems.

1 Introduction

Most control system models in use today incorporate in them explicitly "uncertainties" or "perturbations". These uncertainties may model a number of factors, including: dynamics that are neglected to make the model tractable, as with large scale structures; nonlinearities that are either hard to model or too complicated; and parameters that are not known exactly, either because they are hard to measure or because of varying manufacturing conditions. A widely-used model for uncertain systems, shown in Fig. 1, consists of a nominal finite-dimensional, linear, time-invariant system, with the perturbation or uncertainty Δ in the feedback loop. The signal w represents exogenous inputs, and z represents all outputs of interest. Often additional information about Δ is either known or assumed; common examples are that Δ is diagonal or block-diagonal; sector-bounded memoryless, linear time-invariant or parametric; bounded in norm, passive, etc. The analysis of and design for such control system models is usually referred to as "robust control with structured perturbations".

One of the most fundamental questions concerning the system in Fig. 1 is that of stability: "Is the model stable irrespective of the perturbation Δ, that is, do all solutions of the system equations go to zero, irrespective of Δ?" This is also referred to as the robust stability problem. Some of the approaches for solving this problem, with various assumptions on Δ, are the use of the small-gain, passivity or circle-criteria [9], the Popov criterion[17], and μ or K_m analysis methods [10, 19, 7].

Robust stability is but one desired feature of an uncertain system; of considerable importance are questions beyond stability, known broadly as *robust performance* problems. Robust performance analysis problems concern the computation of the worst possible value, over all uncertainties, of performance indices; these performance indices may be bounds on some state variables, norms of the map from w to z etc. An example of a robust performance analysis problem is the so-called $\mathbf{H}_2$ problem, which is the computation of the largest possible RMS value of the output z over all Δ, when the input w is unit-intensity white noise. This finds application where the average value of a certain signal is of interest, when the system is affected by an unpredictable input that is modeled as white noise. Another example is the computation of the largest possible RMS gain, over all uncertainties, from w to z; this is also known as the $\mathbf{H}_\infty$ performance analysis problem.

It has been found recently that several stability analysis methods for control systems can be unified in the setting of integral quadratic constraints with multipliers [18, 5, 16, 20, 14, 15]. As a consequence, several sufficient conditions for robust stability can be performed without frequency sampling, using convex optimization techniques based on linear matrix inequalities [6, 5]. Moreover, multiplier or IQC-based techniques yield a *convex* parametrization of a subset of Lyapunov functions that prove robust stability [4]; this is of central importance to this paper. By imposing additional conditions on these Lyapunov functions, we will derive bounds on robust performance (see [8] for numerous examples) of the system in Fig. 1. The "best" performance bounds can then obtained by numerically optimizing over the subset of Lyapunov functions that prove robust performance. In §2, we describe the general idea behind the approach; we also address here three specific robust performance analysis problems using this general approach. In §3, we present a simple numerical example, and conclude with §4.

2 Robust performance bounds from Lyapunov functions

Consider the uncertain system shown in Fig. 1, consisting of a stable finite-dimensional linear system with an uncertainty or perturbation Δ in the feedback loop:

$$\frac{d}{dt}x(t) = Ax(t) + B_p p(t) + B_w w(t), \quad q(t) = C_q x(t) + D_{qp} p(t), \quad z(t) = C_z x(t), \tag{1a}$$

$$p(t) = -\Delta(q, t). \tag{1b}$$

where $x(t) \in \mathbb{R}^n$, $p(t) \in \mathbb{R}^{n_p}$, $q(t) \in \mathbb{R}^{n_q}$, $w(t) \in \mathbb{R}^{n_w}$ and $z(t) \in \mathbb{R}^{n_z}$. (For convenience, we have assumed that there is no feed-through from w to z, w to q and p to z.) We shall assume that $n_p = n_q = m$; with little difficulty, the results herein can be extended to the more general case $n_p \neq n_q$. For future reference, we let $G(s)$ denote the transfer function of the linear part of the system, from $[w^T \quad p^T]^T$ to $[z^T \quad q^T]^T$. Equations (1) can be interpreted as representing a family of systems, each member corresponding to some Δ. The perturbation Δ is in general nonlinear. Additional information about the structure (diagonal, block-diagonal etc) and nature (nonlinear bounded, sector-bounded, LTI, parametric, etc) is usually available.

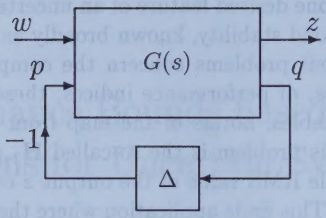


Figure 1: Robustness analysis framework.

Suppose there exists a positive-definite function[1] $V : \mathbb{R}^n \times \mathbb{R}_+ \to \mathbb{R}_+$ such that $dV(x,t)/dt < 0$ along the trajectories of (1). Then from standard arguments from Lyapunov theory, the system (1) is robustly stable, i.e., for every member of the family and for every initial condition, every solution $x(t)$ of (1) satisfies $\lim_{t\to\infty} x(t) = 0$.

Additional conditions on the Lyapunov function can be imposed, yielding robust performance bounds as follows (see for example [3, §6] and [8]): For a given initial condition $x(0)$ for the linear part (1a), suppose that the Lyapunov function V satisfies

$$\frac{d}{dt}V(x,t) < -\phi_\alpha(x(t), w(t), z(t)) \text{ along the trajectories of (1).} \tag{2}$$

Then it is a simple exercise to show that

$$V(x(0),0) > \int_0^\infty \phi_\alpha(x(t), w(t), z(t))\, dt, \tag{3}$$

or we have an upper bound on the performance index

$$\mathcal{P}_\alpha \triangleq \int_0^\infty \phi_\alpha(x(t), w(t), z(t))\, dt. \tag{4}$$

Thus, if we can find a Lyapunov function V satisfying (2), then we can obtain upper bounds on robust performance indices of the form (4), for system (1). Natural questions then are:

1. What is the form of these Lyapunov functions (quadratic function of $x(t)$, some functional of $x(t)$, etc)?
2. How can we optimize over the set of such Lyapunov functions, in order to get the best bounds on robust performance?

The answer to the first question, as we will see through the examples presented in §2.1–§2.3, depends on the structure and nature of Δ. A partial answer to the second question is provided by numerical optimization based on linear matrix inequalities (LMIs)[2]. In particular, we will show that in several cases, a rich subset of Lyapunov functions that satisfy condition (2) can be parametrized using LMIs, and that the objective $V(x(0),0)$ is a convex function of the parameters. Therefore performance bounds for the system (1) can be very efficiently computed using standard software for numerical LMI optimization [12, 13, 25].

[1] For precise technical definitions, mathematical preliminaries, and the notation used here, see [9] and [21].

[2] We refer the reader to the book [8] for an introduction to the application of LMI optimization to problems from system and control theory.

2.1 Example: Unstructured strictly passive Δ; bound on the output energy

Consider the special case when the uncertainty Δ is nonlinear and strictly passive, i.e., for some real β and $\epsilon > 0$,

$$\int_0^T q(t)^T \Delta(q,t)\, dt \geq \epsilon \int_0^T q(t)^T q(t)\, dt - \beta \text{ for all } T \geq 0 \text{ and all } q \in \mathbf{L}_2. \tag{5}$$

The performance analysis problem considered here is the following:

> With exogenous input $w = 0$, find bounds on the energy in the output, i.e., $\int_0^\infty z(t)^T z(t)\, dt$, for a given initial condition $x(0)$ of the state of the linear part of the system.

In order to derive upper bounds on $\int_0^\infty z(t)^T z(t)\, dt$, we seek Lyapunov functions that satisfy condition (2) with $\phi_\alpha(x,w,z) = z^T z$, i.e.,

$$\frac{d}{dt} V(x,t) < -z(t)^T z(t). \tag{6}$$

Then $V(x(0),0)$ yields an upper bound on the worst-case output energy.

It turns out (see [22, 23]) that the Lyapunov functions that are appropriate here are of the form

$$V(x,t) = x(t)^T P x(t) - 2\int_0^t p(\tau)^T q(\tau)\, d\tau + 2\beta,$$

where $P > 0$. Condition (6) holds along the trajectories of system (1) if the following condition holds:

$$\begin{bmatrix} A^T P + PA + C_z^T C_z & PB_p - C_q^T \\ B_p^T P - C_q & -(D_{qp} + D_{qp}^T) \end{bmatrix} < 0. \tag{7}$$

Thus, the best upper bound on the worst-case energy of z, for a given initial condition $x(0)$ of the linear part of the system, is simply

$$\begin{aligned} &\text{minimize} && x(0)^T P x(0) \\ &\text{subject to} && P > 0, \text{ and } (7) \end{aligned}$$

2.2 Example: Diagonal, time-invariant, sector-bounded, memoryless nonlinearities Δ; bound on the state variables

Next, consider the case when: D_{qp} is zero; the uncertainty Δ is diagonal, i.e., $p_i(t) = -\delta_i(q_i(t))$; and each δ_i is a time-invariant, memoryless nonlinearity in sector $[0, \infty)$. For this case, the performance analysis problem considered is:

> With exogenous input $w = 0$, find bounds on the state $x(t)$, $t \geq 0$, for a given initial condition $x(0)$ of the state of the linear part of the system.

In other words, we seek an invariant set for the trajectories $x(t)$ of the linear part of the system. For every positive-definite function V for which $dV(x,t)/dt < 0$ holds along the trajectories of system (1), we have

$$V(x(t),t) \leq V(x(0),0) \text{ for } t \geq 0.$$

This inequality can then be used to derive bounds on $x(t)$.

It turns out (see [9, 8]) that the Lyapunov functions that are appropriate here are of the form

$$V(x,t) = x(t)^T P x(t) + 2\sum_{i=1}^{m} \mu_i \int_0^{q_i(t)} \delta_i(\sigma)\, d\sigma - 2\sum_{i=1}^{m} \lambda_i \int_0^t p_i(\tau) q_i(\tau)\, d\tau,$$

where $P > 0$, $\mu_i \geq 0$ and $\lambda_i \geq 0$. In this case, condition $dV(x,t)/dt < 0$ is equivalent to the LMI with

$$\begin{bmatrix} A^T P + PA & PB_p - (\Lambda C_q + MC_qA)^T \\ B_p^T P - \Lambda C_q - MC_qA & -(MC_qB_p + B_p^T C_q^T M) \end{bmatrix} < 0 \tag{8}$$

where $P > 0$, $M = \mathbf{diag}\,(\mu_1, \ldots, \mu_m) \geq 0$ and $\Lambda = \mathbf{diag}\,(\lambda_1, \ldots, \lambda_m) \geq 0$. If condition (8) holds, then the ellipsoid

$$\mathcal{E} = \left\{ \psi \in \mathbb{R}^n \;\middle|\; \psi^T P \psi \leq x(0)^T P x(0) \right\}$$

is an invariant ellipsoid for the state x.

The problem of finding the smallest invariant ellipsoid (using our techniques), i.e., one with the smallest major axis, can be solved by solving the LMI

$$\begin{aligned} &\text{maximize} && \mu \\ &\text{subject to} && P > \mu I, \;\; x(0)^T P x(0) \leq 1 \text{ and } (8) \end{aligned}$$

and using the optimal P to construct the invariant ellipsoid.

2.3 Example: Diagonal, constant, real Δ; bound on the largest $\mathbf{H}_2$ norm

We finally consider the case when Δ is a constant, unknown, diagonal matrix, with entries that are nonnegative. The performance analysis problem considered is:

Find the worst-case $\mathbf{H}_2$ norm from w to z over all Δ.

The Lyapunov functions that are appropriate in this case are motivated by multiplier theory [9, 5, 4]. Consider the system shown in Fig. 2, where

1. $W_+(s)$, $W_-(-s)^T$ are diagonal, stable and stably invertible
2. $W_-(-s)^T \Delta W_+(s)^{-1}$ is strictly passive for all Δ (9)

Then, there is a one-to-one correspondence between the state-trajectories of the systems in Figs. 1 and 2. We have translated our knowledge of Δ into extra degrees of

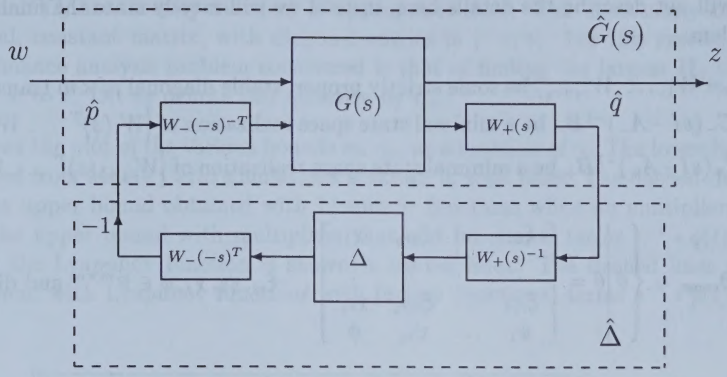


Figure 2: System with multipliers.

freedom, represented by our multipliers W_+ and W_-. Then, we can search for Lyapunov functions for the "augmented" system, which in turn, yields structured Lyapunov functionals for the original system (see [4]).

With $\hat{x}$ denoting the state vector of $\hat{G}$, let the state-equations of the system in Fig. 2 be

$$\frac{d}{dt}\hat{x}(t) = \hat{A}\hat{x}(t) + \hat{B}_p\hat{p}(t) + \hat{B}_w w(t), \quad \hat{q}(t) = \hat{C}_q\hat{x}(t) + \hat{D}_{qp}\hat{p}(t), \quad z(t) = \hat{C}_z\hat{x}(t), \quad \text{(10a)}$$

$$\hat{p}(t) = -\hat{\Delta}(\hat{q}(t), t) \quad \text{(10b)}$$

Then, a suitable Lyapunov function is

$$V(\hat{x}, t) = \hat{x}(t)^T\hat{P}\hat{x}(t) - 2\int_0^t \hat{p}(\tau)^T\hat{q}(\tau)\,d\tau,$$

where $\hat{P} > 0$.

It can be shown that if this Lyapunov function satisfies

$$\frac{d}{dt}V(\hat{x}, t) < -z(t)^T z(t), \quad \text{(11)}$$

then $\sqrt{\mathbf{Tr}\,\hat{B}_w^T\hat{P}\hat{B}_w}$ yields an upper bound on the worst-case $\mathbf{H}_2$ norm from w to z over all Δ.

The problem of computing the best upper bound then is the following:

$$\begin{array}{ll} \text{minimize} & \sqrt{\mathbf{Tr}\,\hat{B}_w^T\hat{P}\hat{B}_w} \\ \text{subject to} & \hat{P} > 0, \quad (9) \text{ and } (11) \end{array}$$

The variables in this optimization problem are $\hat{P}$, and appropriate W_+ and W_-; owing to the latter two variables, it is an infinite-dimensional optimization problem. In practice, it can be approximately solved by restricting W_+ and W_- to lie in a finite-dimensional set.

We will not describe the details here, instead we will merely state the final optimization problem:

Let $W_1, \ldots, W_{r_1+r_2}$, be some strictly proper, stable diagonal $m \times m$ transfer functions. Let $C_-(sI-A_-)^{-1}B_-$ be a minimal state space realization of $\left[W_1(s)^T \ \ldots \ W_{r_1}(s)^T\right]^T$, and let $C_+(sI-A_+)^{-1}B_+$ be a minimal state space realization of $\left[W_{r_1+1}(s)^T \ \ldots \ W_{r_1+r_2}(s)^T\right]^T$. Let

$$\mathbf{\Theta}_{\text{const}} = \left\{ \theta \;\middle|\; \theta = \begin{bmatrix} \xi_{11} & \cdots & \xi_{1r_2} & \chi_1 \\ \vdots & \ddots & \vdots & \vdots \\ \xi_{r_1 1} & \cdots & \xi_{r_1 r_2} & \chi_{r_1} \\ \psi_1 & \cdots & \psi_{r_2} & \phi \end{bmatrix}, \quad \xi_{ij}, \psi_i, \chi_j, \phi \in \mathbb{R}^{m\times m} \text{ and diagonal} \right\}.$$

Then an upper bound to the worst-case $\mathbf{H}_2$ norm from w to z for system (1) when $\Delta > 0$ is diagonal and real is given by the square root of the optimal objective of the following LMI problem:

Minimize: $\quad \mathbf{Tr}\, \tilde{B}_w^T \tilde{P} \tilde{B}_w$

subject to :

$$\begin{bmatrix} A_{\mathrm{W}}^T P_{\mathrm{W}} + P_{\mathrm{W}} A_{\mathrm{W}} & P_{\mathrm{W}} B_{\mathrm{W}} \\ B_{\mathrm{W}}^T P_{\mathrm{W}} & 0 \end{bmatrix} - \begin{bmatrix} C_{\mathrm{W}} & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} 0 & \theta_{11}^T & \theta_{21}^T \\ \theta_{11} & 0 & \theta_{12} \\ \theta_{21} & \theta_{12}^T & \theta_{22}+\theta_{22}^T \end{bmatrix} \begin{bmatrix} C_{\mathrm{W}} & 0 \\ 0 & I \end{bmatrix} \leq 0$$

$$\begin{bmatrix} \tilde{A}^T\tilde{P} + \tilde{P}\tilde{A} + \tilde{C}_z^T \tilde{C}_z & \tilde{P}\tilde{B}_p \\ \tilde{B}_p^T \tilde{P} & 0 \end{bmatrix} - \begin{bmatrix} \tilde{C}_q & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} 0 & \theta_{11} & \theta_{12} & \theta_{12} D_{qp} \\ \theta_{11}^T & 0 & 0 & \theta_{21}^T \\ \theta_{12}^T & 0 & 0 & \theta_{22}^T \\ D_{qp}^T \theta_{12}^T & \theta_{21} & \theta_{22} & \theta_{22} D_{qp} + D_{qp}^T \theta_{22}^T \end{bmatrix} \begin{bmatrix} \tilde{C}_q & 0 \\ 0 & I \end{bmatrix} \leq 0$$

$$\begin{bmatrix} \theta_{11} & \theta_{12} \\ \theta_{21} & \theta_{22} \end{bmatrix} \in \mathbf{\Theta}_{\text{const}}, \quad \tilde{P} > 0, \quad P_{\mathrm{W}} > 0$$

where

$$A_{\mathrm{W}} = \begin{bmatrix} A_+ & 0 \\ 0 & A_- \end{bmatrix}, \quad B_{\mathrm{W}} = \begin{bmatrix} B_+ \\ B_- \end{bmatrix}, \quad C_{\mathrm{W}} = \begin{bmatrix} C_+ & 0 \\ 0 & C_- \end{bmatrix},$$

$$\tilde{A} = \begin{bmatrix} A_- & 0 & 0 \\ 0 & A_+ & B_+ C_q \\ 0 & 0 & A \end{bmatrix}, \quad \tilde{B}_p = \begin{bmatrix} B_- \\ B_+ D_{qp} \\ B_p \end{bmatrix}, \quad \tilde{C}_q = \begin{bmatrix} C_- & 0 & 0 \\ 0 & C_+ & 0 \\ 0 & 0 & C_q \end{bmatrix},$$

and

$$\tilde{B}_w = \begin{bmatrix} 0 \\ 0 \\ B_w \end{bmatrix}, \quad \text{and} \quad \tilde{C}_z = \begin{bmatrix} 0 & 0 & C_z \end{bmatrix}.$$

3 A numerical example

Consider a specific example of system (1), where

$$A = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix}, \ B_p = \begin{bmatrix} 10 & 0 \\ 0 & 2/5 \end{bmatrix}, \ D_{qp} = \begin{bmatrix} 2 & -2 \\ -2 & 2 \end{bmatrix},$$

and C_q, B_w and C_z are identity matrices of appropriate sizes. The uncertainty Δ is a 2×2 diagonal, constant matrix, with diagonal entries in $[-\gamma, \gamma]$. For this system, the robust performance analysis problem considered is that of finding the largest $\mathbf{H}_2$ norm, over Δ, from w to z. Let us denote this quantity by η_{wc}.

Fig. 3 shows the plot of the various bounds on η_{wc} as a function of γ. The lower bound on η_{wc} obtained from Monte Carlo simulations is shown in solid lines. The dashed-dotted lines show the upper bound obtained with Lyapunov functions when no multipliers are used, while the upper bound with multipliers that add functional terms $e^{-t} * q(t)$ and $te^{-t} * q(t)$ to the Lyapunov function is shown in dotted lines. The dashed lines show the upper bound with Lyapunov functions with further functional terms $e^{-t} * p(t)$ and $te^{-t} * p(t)$.

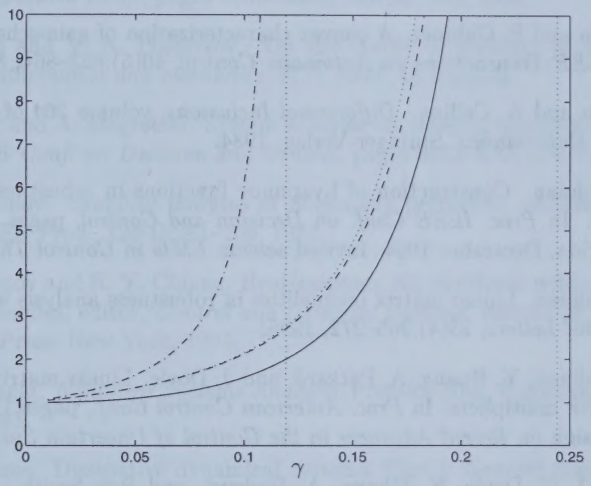


Figure 3: Plot of the bounds on the worst-case $\mathbf{H}_2$ norm as a function of γ.

Given the simplicity of this example, the Monte Carlo runs can be thought of yielding the exact value of η_{wc} with high likelihood. Then, it is evident from the plots that the upper bound on η_{wc}, obtained without any multipliers diverges from the actual value of η_{wc} rather rapidly as γ increases. Around $\gamma = 0.12$, this upper bound tends to infinity; the reason is that even robust stability cannot be established without any multipliers for $\gamma > 0.12$. The upper bound from Lyapunov functions for the augmented systems however, appear to be fairly tight in this example. As expected, the addition of additional dynamics leads to an improvement in the upper bound.

4 Conclusion

The techniques presented here provide a framework for generating Lyapunov functions for systems affected by structured nonlinearities and perturbations. The ideas presented extend small-gain, passivity, Popov, μ and real-μ stability tests to computing robust performance bounds. The underlying numerical methods do no require frequency sampling;

instead, the methods are based on state-space techniques and convex LMI optimization, and yield guaranteed bounds. However, the form of the Lyapunov function has to decided on, based on the information known about the uncertainty. Also, the question of how conservative the bounds obtained are, remains. The general framework presented here offers the potential for control synthesis to optimize robust performance bounds, using a "gain-scheduled" framework [24, 11, 2, 1].

References

[1] P. Apkarian, J-M. Biannic, and P. Gahinet. Self-scheduled $\mathbf{H}_\infty$ control of missile via linear matrix inequalities. *J. Guidance and Control,*, 18(3):532–538, May–Jun 1995.

[2] P. Apkarian and P. Gahinet. A convex characterization of gain-scheduled H_∞ controllers. *IEEE Transactions on Automatic Control*, 40(5):853–864, May 1995.

[3] J. P. Aubin and A. Cellina. *Differential Inclusions*, volume 264 of *Comprehensive Studies in Mathematics*. Springer-Verlag, 1984.

[4] V. Balakrishnan. Construction of Lyapunov functions in robustness analysis with multipliers. In *Proc. IEEE Conf. on Decision and Control*, pages 2021–2025, Orlando, Florida, December 1994. Invited session *LMIs in Control Theory*.

[5] V. Balakrishnan. Linear matrix inequalities in robustness analysis with multipliers. *Syst. Control Letters*, 25(4):265–272, 1995.

[6] V. Balakrishnan, Y. Huang, A. Packard, and J. Doyle. Linear matrix inequalities in analysis with multipliers. In *Proc. American Control Conf.*, pages 1228–1232, 1994. Invited session on *Recent Advances in the Control of Uncertain Systems*.

[7] G. Balas, J. C. Doyle, K. Glover, A. Packard, and Roy Smith. *μ-analysis and synthesis*. MUSYN, inc., and The Mathworks, Inc., 1991.

[8] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory*, volume 15 of *Studies in Applied Mathematics*. SIAM, Philadelphia, PA, June 1994.

[9] C. A. Desoer and M. Vidyasagar. *Feedback Systems: Input-Output Properties*. Academic Press, New York, 1975.

[10] J. Doyle. Analysis of feedback systems with structured uncertainties. *IEE Proc.*, 129-D(6):242–250, November 1982.

[11] P. Gahinet, P. Apkarian, and M. Chilali. Affine parameter-dependent Lyapunov functions for real parameter uncertainty. *IEEE Trans. Aut. Control*, 41(3):436–442, March 1996.

[12] P. Gahinet and A. Nemirovskii. *LMI Control Toolbox: the LMI Lab*. The MathWorks, Inc., 1995.

[13] L. El Ghaoui, F. Delebecque, and R. Nikoukhah. *LMITOOL: A User-friendly Interface for LMI Optimization.* ENSTA/INRIA, 1995. Software available via anonymous FTP from `ftp.inria.fr`, under directory `pub/elghaoui/lmitool`.

[14] K.-C. Goh and M. G. Safonov. Robust analysis, sectors, and quadratic functionals. In *Proc. IEEE Conf. on Decision and Control*, pages 1988–1993, New Orleans, LA, 1995.

[15] J. P. How and S. R. Hall. Connections between the Popov stability criterion and bounds for real parametric uncertainty. In *Proc. American Control Conf.*, volume 2, pages 1084–1089, June 1993.

[16] U. Jonsson and A. Rantzer. Unifying format for multiplier optimization. In *Proc. American Control Conf.*, pages 3859–3860, Seattle, WA, 1995.

[17] A. I. Lur'e and V. N. Postnikov. On the theory of stability of control systems. *Applied mathematics and mechanics*, 8(3), 1944. In Russian.

[18] A. Rantzer and A. Megretski. System analysis via integral quadratic constraints. In *Proc. IEEE Conf. on Decision and Control*, pages 3062–3067, December 1994.

[19] M. G. Safonov. Stability margins of diagonally perturbed multivariable feedback systems. *IEE Proc.*, 129-D:251–256, 1982.

[20] M. G. Safonov and R. Y. Chiang. Real/complex K_m-synthesis without curve fitting. In C. T. Leondes, editor, *Control and Dynamic Systems*, volume 56, pages 303–324. Academic Press, New York, 1993.

[21] M. Vidyasagar. *Nonlinear Systems Analysis.* Prentice-Hall, Englewood Cliffs, second edition, 1993.

[22] J. C. Willems. Dissipative dynamical systems, Part I: General theory. *Archive for Rational Mechanics and Analysis*, 45:321–351, 1972.

[23] J. C. Willems. Dissipative dynamical systems, Part II: Linear systems with quadratic supply rates. *Archive for Rational Mechanics and Analysis*, 45:352–393, 1972.

[24] F. Wu, X. H. Yang, A. Packard, and G. Becker. Induced $\mathbf{L}_2$norm control for LPV system with bounded parameter variation rates. In *Proc. American Control Conf.*, pages 2379–2383, Seattle, WA, 1995.

[25] S.-P. Wu and S. Boyd. SDPSOL: *A Parser/Solver for Semidefinite Programming and Determinant Maximization Problems with Matrix Structure. User's Guide, Version Beta.* Stanford University, June 1996.

A NEW FORMULATION OF SOME DISCRETE-TIME STOCHASTIC-PARAMETER STATE ESTIMATION PROBLEMS *

Edwin E. Yaz, Department of Electrical Engineering
University of Arkansas, Fayetteville, AR 72701
Yvonne I. Yaz, Department of Mathematics
Centenary College, Shreveport, LA 71134

Abstract

Linear unbiased full-order state estimation problem for discrete-time models with stochastic parameters and additive finite energy type disturbance signals is reformulated in terms of linear matrix inequalities. Two estimation problems that are considered are the design for mean-square bounded estimation error and the design for the mean-square stochastic version of the suboptimal H_∞ estimator. These two designs are shown to apply to both the estimation with random sensor delay and estimation under observation uncertainty.

Keywords: State estimation; Stochastic-parameter systems; Linear matrix inequalities; Discrete-time systems; Sensor delays.

Introduction

State estimator design problem is reformulated using linear matrix inequalities (LMIs) for linear discrete-time state space models with white stochastic parameters. This is done as an alternative to the conventional minimum variance linear filtering approach used in this field [1]. The purpose behind this is to utilize the efficient numerical schemes proposed for the solution of LMIs. In the model used, the process and the measurements are assumed to be affected by additive white noises with finite energy. Linear, constant gain, unbiased, and full-order (identity) estimators are sought. Two main estimation problems are tackled: the design aiming at mean-square bounded estimation error and the mean-square stochastic version of the suboptimal H_∞ estimator design. The general result obtained is applied to two problems: i) estimation with random sensor-delay, and ii) estimation under observation uncertainty where the measurement either contains the signal or is composed of noise only.

The following notation is used. For an n-dimensional vector of real elements $x \in \mathbf{R}^n$, $\| x \|$ denotes the 2-norm, $(x^T x)^{1/2}$. For an $n \times n$ symmetric matrix A with real elements $A \in \mathbf{R}^{n \times n}$, $A > 0$ ($A < 0$), $\lambda_{\max}(A)$ ($\lambda_{\min}(A)$) respectively mean positive definite (negative definite) A and its maximum (minimum) eigenvalue. $E\{x\}$ and $E\{x/y\}$ respectively denote the expected value of x and its expected value conditional on y. Also, $E\{x_k\}$ is written as $E\{x\}$ and $E\{x_k x_k{}^T\}$ is written as $E\{xx^T\}$ for a weakly stationary sequence x_k.

Some Basic Results

The following is an auxiliary result that will allow us to prove a general theorem for state estimator designs for different models.

* This work was partially supported by NSF under grant ECS-9922798 and Centenary College Faculty Summer Research Grant.

Lemma 1. Consider the linear stochastic-parameter discrete-time system

$$x_{k+1} = Ax_k + Bw_k \tag{1}$$

$$y_k = Cx_k + Dw_k \tag{2}$$

where $x_k \in \mathbf{R}^n$ is the state, $y_k \in \mathbf{R}^p$ is the output and $w_k \in \mathbf{R}^m$ is a stochastic l_2-type signal with

$$\sum_{k=0}^{\infty} E\{\| w_k \|^2\} \leq 1 \tag{3}$$

which is white or $E\{w_k w_j{}^T\} = 0$ for all $k \neq j \geq 0$. The elements of the parameter matrices A_k and C_k are second order, weakly stationary, white noise sequences and B and D are constant matrices. The sequences $\{A_k,\ C_k\}$, $\{w_k\}$ for all $k \geq 0$ and x_0 are assumed to be mutually uncorrelated. Then the following are true:
i) If there exists a matrix $P > 0$ and a scalar $\gamma > 0$ such that

$$\begin{pmatrix} P & 0 \\ 0 & \gamma I \end{pmatrix} - \begin{pmatrix} E\{A^T P A\} & E\{A^T\} P B \\ B^T P E\{A\} & B^T P B \end{pmatrix} > 0 \tag{4}$$

then the weighted second moment of the state satisfies

$$E\{x_N{}^T P x_N\} < E\{x_0{}^T P x_0\} + \gamma \tag{5}$$

for every $N > 0$.
ii) If there exists a matrix $P > 0$ and a scalar $\gamma > 0$ such that

$$\begin{pmatrix} P & 0 \\ 0 & \gamma I \end{pmatrix} - \begin{pmatrix} E\{A^T P A\} & E\{A^T\} P B \\ B^T P E\{A\} & B^T P B \end{pmatrix} - \begin{pmatrix} E\{C^T C\} & E\{C^T\} D \\ D^T E\{C\} & D^T D \end{pmatrix} > 0 \tag{6}$$

then the output energy satisfies

$$\sum_{k=0}^{N} E\{\| y_k \|\}^2 < E\{x_0{}^T P x_0\} + \gamma \tag{7}$$

for every $N > 0$.

Note that the above matrix inequalities (4) and (6) are both linear in P and γ.

Proof. It is done by defining $V_k = x_k{}^T P x_k$. First it is shown that (4) implies

$$E\{ V_{k+1} / x_k, w_k\} - V_k - \gamma \| u_k \|^2 < 0$$

which yields

$$\sum_{k=0}^{N-1} (E\{ V_{k+1}\} - E\{ V_k\}) < \gamma \sum_{k=0}^{N-1} E\{\| w_k \|^2\}$$

This is obtained by taking expectation of both sides and using the interlacing property of expectations. Simplifying the left hand summation and using (3) on the right hand side gives (5). Second, it is shown that (6) implies

$$E\{V_{k+1}/x_k, w_k\} - V_k + E\{\| y_k \|^2 /x_k, w_k\} - \gamma \| w_k \|^2 < 0$$

or proceeding the same way as before

$$\sum_{k=0}^{N} (E\{V_{k+1}\} - E\{V_k\}) < -\sum_{k=0}^{N} E\{\| y_k \|^2\} + \gamma \sum_{k=0}^{N} E\{\| w_k \|^2\}$$

Simplification yields

$$\sum_{k=0}^{N} E\{\| y_k \|^2\} < E\{V_0\} - E\{V_{N+1}\} + \gamma < E\{V_0\} + \gamma$$

that gives (7).

Now let us consider the system (1) and measurement model (2) again. Suppose, a linear unbiased full-order state estimator (one-step-ahead predictor) is used:

$$\hat{x}_{k+1} = E\{A\}\hat{x}_k + K(y_k - E\{C\}\hat{x}_k), \; x_0 = E\{x_0\} \tag{8}$$

This leads to the following estimation error time evolution:

$$e_{k+1} = (E\{A\} - KE\{C\})e_k + (\tilde{A}_k - K\tilde{C}_k)x_k + (B - KD)w_k \tag{9}$$

where $\tilde{A}_k = A_k - E\{A\}$ and similarly for $\tilde{C}_k$. The main estimation result can be summed up as follows:

Theorem 1. Let the following LMI hold for some matrix $X > 0$ and scalar $\delta > 0$

$$\begin{pmatrix} X - I - E\{A^T X A\} & -E\{A^T\} X B \\ -BXE\{A\} & \delta I - B^T X B \end{pmatrix} > 0 \tag{10}$$

Then, if there exist matrices $P > 0$, Y and a scalar $\gamma > 0$ such that

$$\begin{pmatrix} P & 0 & E\{A\}P - E\{C^T\}Y^T \\ 0 & \gamma I & B^T P - D^T Y^T \\ PE\{A\} - YE\{C\} & PB - YD & P \end{pmatrix} > 0 \tag{11}$$

then with K being found by $K = P^{-1}Y$, we have

$$E\{e_N{}^T P e_N\} < \infty \tag{12}$$

for all $N \geq 0$. If, instead of (11), the following is true

$$\begin{pmatrix} P - \psi^T \psi & 0 & E\{A\}P - E\{C^T\}Y^T \\ 0 & \gamma I & B^T P - D^T Y^T \\ PE\{A\} - YE\{C\} & PB - YD & P \end{pmatrix} > 0 \tag{13}$$

then the performance output defined as

$$z_k = \psi e_k \tag{14}$$

satisfies the energy bound

$$\sum_{k=0}^{N} E\{\| z_k \|^2\} < \infty \tag{15}$$

with K given by $K = P^{-1} Y$.

The exact bounds in (12) and (15) will be developed in the proof.

Proof. Let the LMI (10) hold for $X > 0$ and $\delta > 0$. This guarantees by lemma 1 that

$$\sum_{k=0}^{N} E\{\| x_k \|^2\} < E\{x_0{}^T X x_0\} + \delta \tag{16}$$

for any $N > 0$. Now, for $V_k = e_k{}^T P e_k > 0$, consider

$$E\{ V_{k+1}/e_k,\ w_k\} - V_k - \gamma \| w_k \|^2 \tag{17}$$

which upon substitution from (9) yields

$$\begin{aligned}
&(e_k{}^T \quad w_k{}^T) \\
&\times \begin{pmatrix} (E\{A\} - KE\{C\})^T P (E\{A\} - KE\{C\}) - P & (E\{A\} - KE\{C\})^T) P(B - KD) \\ (B - KD)^T P (E\{A\} - KE\{C\}) & -\gamma I + (B - KD)^T P (B - KD) \end{pmatrix} \\
&\times \begin{pmatrix} e_k \\ w_k \end{pmatrix} + E\{x_k{}^T (\tilde{A} - K\tilde{C})^T P (\tilde{A} - K\tilde{C}) x_k\}
\end{aligned} \tag{18}$$

Note that if (11) holds, then, for $Y = PK$, due to the Schur complement result, which is, the equivalence of

$$\begin{pmatrix} A & B \\ B^T & C \end{pmatrix} > 0$$

and $A - BC^{-1}B^T > 0$ with $C > 0$, the first term in (18) is negative definite. Let us denote (18) as

$$-(e_k{}^T \quad w_k{}^T) M_1(P,\ Y,\ \gamma) \begin{pmatrix} e_k \\ w_k \end{pmatrix} + E\{x_k{}^T \tilde{M}(P, K) x_k\} \tag{19}$$

Due to the second order nature of the stochastic parameters and using Rayleigh's inequality, we obtain

$$\begin{aligned}
E\{ V_{k+1}/e_k,\ w_k\} - V_k - \gamma \| w_k \|^2 = &-(e_k{}^T \quad w_k{}^T) M_1(P,\ Y,\ \gamma) \begin{pmatrix} e_k \\ w_k \end{pmatrix} \\
&+ \lambda_{max}(\tilde{M}(P,\ K)) E\{\| x_k \|^2\}
\end{aligned} \tag{20}$$

Taking the expected value of both sides, using the property of nested expectations, and summing over k, (20) yields

$$\sum_{k=0}^{N-1} E\{ V_{k+1} - V_k\} < \gamma \sum_{k=0}^{N-1} E\{\| w_k \|^2\} + \lambda_{max}(\tilde{M}(P,\ K)) \sum_{k=0}^{N-1} E\{\| x_k \|^2\} \tag{21}$$

or

$$V_N < E\{ V_0\} + \gamma + \lambda_{max}(\tilde{M}(P,\ K))[E\{{x_0}^T X x_0\} + \delta] \tag{22}$$

which directly implies (12). The fact that (13) implies (15) can similarly be shown with the addition of $\| z_k \|^2$ where z_k is given in (14). This is used in (17) and going through the same procedure, yields

$$\sum_{k=0}^{N} E\{\| z_k \|^2\} < E\{ V_0\} + \gamma + \lambda_{max}(\tilde{M}(P,\ K))[E\{{x_0}^T X x_0\} + \delta] \tag{23}$$

This completes the proof.

Note that LMI (10) implies, via the Schur complement result, that

$$X > E\{A^T X A\} + I > E\{A^T X A\}$$

which means that the unforced system

$$x_{k+1} = A_k x_k$$

is necessarily mean-square exponentially stable [1]. This was found to be true for application of steady state Kalman filters in [2].

In the next two sections, the application of these basic results to two important estimation problems will be discussed.

Random Delay in the Sensor

The first application that will be considered is the problem of state estimation with a random sensor-delay. A slightly modified form (which is taken to fit the description in Theorem 1 of the system dynamics and delayed sensor outputs that are modeled in [3], [4] is as follows:

$${x_{k+1}}' = \Phi {x_k}' + F w_k \tag{24}$$

$${y_k}' = C' {x_k}' \tag{25}$$

$$y_k = (1 - \xi_k) {y_k}' + \xi_k {y_{k-1}}' \tag{26}$$

w_k is a zero mean white noise satisfying (3). The binary white noise sequence ξ_k which is independent of ${x_0}'$ and w_k for $k > 0$, takes only values "0" and "1" corresponding to no

delay and one-step delay case, respectively. It has the statistics $E\{\xi_k\} = \alpha = E\{\xi_k{}^2\}$ with $\alpha \in [0, 1]$. The above description can be more compactly written by defining

$$x_k := \begin{pmatrix} x_k{}' \\ x'_{k-1} \end{pmatrix}$$

as

$$x_{k+1} = \begin{pmatrix} \Phi & 0 \\ I_n & 0 \end{pmatrix} x_k + \begin{pmatrix} F \\ 0 \end{pmatrix} w_k := Ax_k + Bw_k \tag{27}$$

$$y_k = [(1 - \xi_k)C', \xi_k C']x_k := C_k x_k \tag{28}$$

Since, we are interested in linear unbiased full-order state estimator (one-step-ahead predictor)

$$\hat{x}_{k+1} = A\hat{x}_k + K(y_k - BE\{C\}\hat{x}_k),\ \hat{x}_0 = E\{x_0\} \tag{29}$$

the state estimation error evolution becomes

$$e_{k+1} = (A - KE\{C\})e_k + Bw_k - K\tilde{C}_k x_k \tag{30}$$

where $\tilde{C}_k = C_k - E\{C\}$. Since only the estimation error for the current state is of interest, we take

$$z_k = (I_n \quad 0)\, x_k := \psi x_k \tag{31}$$

With the identification of the design parameters in (30), Theorem 1 can be used to design state estimators.

Corollary 1. Let $E\{A\} = A$, $E\{A^T X A\} = E\{A^T\} X E\{A\}$, $\psi = (I_n \quad 0)$, and $D = 0$ in (10) – (13). Then, the results of Theorem 1 hold for the estimator (29) applied to the model (24) – (26).

Uncertain Observations

The second application problem that will be considered is the uncertain measurement problem where there is a probability of measuring signal in noise or just noise only. Consider the linear discrete-time system

$$x_{k+1} = Ax_k + Bw_k \tag{32}$$

with the measurement sequence [5] - [6]

$$y_k = \xi_k C_k + Dw_k \tag{33}$$

where w_k is white and satisfies (3), and ξ_k has again the same statistical description as in the previous section with

$$\alpha := \text{prob}\{\xi_k = 1\} \text{ and } (1 - \alpha) := \text{prob}\{\xi_k = 0\} \tag{34}$$

where the realization $\xi_k = 1$ means that the signal is present in the noise corrupted measurement and $\xi_k = 0$ means that what is being measured is just noise. The linear unbiased full-order state estimator is given in [5] and [6] by

$$\hat{x}_{k+1} = A\hat{x}_k + K(y_k - \alpha C\hat{x}_k),\ \hat{x}_0 = E\{x_0\} \tag{35}$$

which yields the state estimation error

$$e_{k+1} = (A - \alpha KC)e_k + Bw_k - K(\tilde{\xi}_k Cx_k + Dw_k) \tag{36}$$

where $\tilde{\xi}_k = \xi_k - \alpha$. Since, any linear combination of state variables may be of interest in estimation, we choose z_k as

$$z_k = \psi x_k \tag{37}$$

where ψ is an arbitrary matrix.

One can easily use Theorem 1 to obtain state estimators as follows:

Corollary 2. Let $E\{A\} = A$, $E\{C\} = \alpha C$, $E\{A^TXA\} = E\{A^T\}XE\{A\}$ in (10) – (13). Then, the results of Theorem 1 hold for the estimator (35) applied to the model (32) – (34).

Conclusion

Linear unbiased state estimation has been considered for discrete-time stochastic parameter models corrupted by finite energy type disturbances. LMIs have been used to pose relevant estimation problems and possible applications to a couple of practical situations are indicated. Since, it follows by Rayleigh's quotient inequality that

$$x_N{}^T P x_N \geq \lambda_{min}(P) x_N{}^T x_N$$

$$\lambda_{\max}(\tilde{M}(P,\ K)) \leq \lambda_{max}(P)\lambda_{max}(E\{(\tilde{A} - K\tilde{C})^T(\tilde{A} - K\tilde{C})\})$$

it is possible to further sharpen these results. For example, it follows from (22) that the bound on the second moment of the estimation error can be reduced by maximizing $\lambda_{min}(P)$ while at the same time minimizing $\lambda_{max}(P)$ and γ subject to the LMIs $P > 0$, $\gamma > 0$ and (11). Similarly, it follows from (23) that the bound on the output energy can be improved by minimizing $\lambda_{max}(P)$ and γ subject to $P > 0$, $\gamma > 0$ and (13). These are classified as eigenvalue problems in [7]. Another numerical improvement can be affected via the use of Finsler's lemma [7] to reduce one of the solution matrices Y in (11) or (13) to a scalar σ. These improvements together with the results of our simulation studies will be reported soon in the literature.

REFERENCES

1. E.Yaz,"Observer design for discrete-time stochastic systems," Control and Dynamic Systems (ed. C.T.Leondes), vol. 73, pp. 121 - 158, Academic Press: San Diego, CA. 1995.
2. W.L.DeKoning, "Optimal estimation of linear discrete-time systems with stochastic parameters," Automatica, vol. 20, pp. 113 - 115, 1984.

3. L.W.Liou and A.Ray,"A stochastic regulator for integrated communication and control systems: Part II - Numerical analysis and simulation," ASME J. Dynamic Systems, vol.113, pp. 612 - 619, 1991.
4. E.Yaz and A.Ray," Linear unbiased state estimation for random models with sensor delay," Proc. of 1996 CDC, Kobe, Japan (to appear).
5. N.E.Nahi," Optimal recursive estimation with uncertain observations," IEEE Trans. Inform. Theory, vol. 15, pp. 457 - 462, 1969.
6. E.Yaz and W.NaNacara," Linear and nonlinear estimation with uncertain observations," Proc. of 1994 ACC, Baltimore, MD, pp.1429 -1433, 1994.
7. S.Boyd, L. El Ghaoui, E.Feron, and V.Balakrishnan, LMIs in System and Control Theory, SIAM: Philadelphia, PA, 1994.

Probabilistic Enhancement of Robustness Margins Provided by Linear Matrix Inequalities†

B. R. Barmish*, C. M. Lagoa* and P. S. Shcherbakov**

Abstract

In this paper, a new paradigm is provided for robustness analysis involving linear matrix inequalities (LMI). There are two salient points to note about the results given here. First, the new framework often leads to robustness measures which are considerably larger than those provided by LMI theory. This is accomplished by introducing an acceptable risk level ε which becomes part of the problem formulation. In other words, we allow for the possibility of "LMI violation" with probability $p = \varepsilon$ and obtain a larger radius of uncertainty in the parameter space. The second salient feature of the theory relates to the fact that robustness practitioners often begin with models for which there is no apriori statistical information about the uncertain parameters; only interval bounds are available. Despite this lack of statistical information, the new paradigm still leads to a readily computable "guaranteed probability" that the performance specification will be satisfied. To this end, in accordance with [4], a special class $\mathcal{F}_r$ of admissible probability distributions plays an important role in the theory. It is shown that distributional robustness with respect to $\mathcal{F}_r$ can be assessed with the aid of uniform sampling.

1. Introduction and New Uncertainty Paradigm

The theory of linear matrix inequalities (LMI) has been used with considerable success to solve large classes of control problems which can be formulated in a rather specific form. Once in the required LMI format, a solution is obtained via convex programming; e.g., see [1] for the theory and [2] for codes associated with computer implementation. The main claim in this paper is simple to state: When LMI theory is used to solve robustness problems with parametric uncertainty, the results which are obtained can be unduly conservative from the point of view of tradeoffs between levels of acceptable risk and the resulting robustness margins. That is, in many cases, a control system designer may be willing to accept a small well-defined level of "risk" in order to obtain larger robustness margins than would be possible with classical LMI theory. To illustrate, if a system specification requires that the H_∞ norm of some uncertain transfer function be bounded by $\gamma > 0$, then our claim is that in many applications, the designer may be satisfied with some suitably small risk probability $p = \varepsilon$. In other words, one accepts this risk that the norm bound is violated in exchange for a 50% increase in the tolerable uncertainty bound.

In the sequel, we describe a new paradigm to study tradeoffs which are precisely of this sort. To this end, the takeoff point for this work is the so-called Uniformity Principle introduced in [3] and [4]. As explained below, the model of uncertainty which we use enables probabilistic analysis without detailed apriori information about the uncertain parameters; i.e., the objective is to deal with uncertainty in a "distribution free" way.

*ECE Department, University of Wisconsin, Madison, WI 53706. ** Moscow Institute of Control Sciences, Moscow, Russia. † Funding for this research was provided by the National Science Foundation under Grant ECS-9418709.

In essence, the theory to follow justifies the use of uniform sampling to estimate both the radius of uncertainty and the associated risk level. Having the sampling distribution in hand, it now becomes possible to use results from Monte Carlo theory to guarantee that reliable estimates can be obtained with computational effort which only increases modestly with the dimension of the uncertainty; e.g., see [5] - [7].

1.1 The Heart of the New Paradigm: In this subsection, we briefly overview the model of uncertainty introduced in [3]. Indeed, the uncertain parameters q_i are viewed as independent random variables with zero mean and given interval bounds $Q_i = [-r_i, r_i]$. The shape of the uncertainty box $Q \doteq Q_1 \times Q_2 \times \cdots \times Q_\ell$ is taken to be fixed but its radius is treated as variable. Therefore, without loss of generality, in the sequel, a scaling can always be introduced to account for "shape" of Q and we work with a single radius $r \geq 0$; i.e., we take $r_i = r$ for $i = 1, \ldots, \ell$ and use Q_r to denote the support set for q. With regard to the density f_i of q_i, it is assumed that $f_i(x_i)$ is symmetric and nonincreasing with respect to $|x_i|$. From an applications point of view, the conditions describing this function are roughly interpreted to mean that small manufacturing errors are more likely than large errors and that positive and negative deviations of q_i from its mean are equally likely. No additional structure for $f_i(x_i)$ is assumed because the intent is to obtain distributionally robust probabilistic estimates. Since the q_i are independent, the vector q has joint density function

$$f(x_1, x_2, \ldots, x_\ell) = f_1(x_1) f_2(x_2) \cdots f_\ell(x_\ell).$$

We write $f_i \in \mathcal{F}_r$ to express admissibility of the density for q_i and denote admissibility of f by writing $f \in \mathcal{F}_r$; in this case the understanding is that $f_i \in \mathcal{F}_r$ for $i = 1, 2, \ldots, \ell$. Given $f \in \mathcal{F}_r$, we introduce notation q_f to denote the random vector having density f. Finally, in order to distinguish the uniform distribution from other members of $\mathcal{F}_r$, we use subscript u_r. Hence, q_{u_r} is the random vector with uniform distribution over Q_r.

To conclude, it is important to provide some motivation for the introduction of the class $\mathcal{F}_r$ of admissible density functions. When working in a robustness framework, we want to make probabilistic assessments of risk without apriori knowledge of the density functions for each q_i. Hence, as already indicated, we want to generate probability estimates which can be viewed as distribution-free. To this end, use of the class $\mathcal{F}_r$ leads to a formulation with associated probability estimate which is guaranteed for every density function $f \in \mathcal{F}_r$.

1.2 Overview: The main objective of this paper is to demonstrate the efficacy of the new paradigm in the context of Linear Matrix Inequalities (LMI). In Section 2, we work at a "high level" in the sense that probabilistic robustness margins are readily defined for very general problems. These margins are distribution-free in the sense of $\mathcal{F}_r$ above and can be viewed as a generalization of classical margins. The notion of *sharp probabilistic estimator* is seen to play an important role. Section 3 concentrates on the particularization of the theory in Section 2 to an LMI context; examples are given showing how an LMI solution can be "amplified" in a probabilistic setting.

Section 4 contains the main result of the paper. Namely, in Theorem 4.2, the description of a sharp estimator is given and it is subsequently shown how this estimator is computed. The proof of the main result is presented in Section 5 and Section 6 is devoted to a numerical example. A major highlight of these examples is the fact that probabilistic margins turn out to be much larger than their deterministic counterparts while only involving low levels of risk. Conclusions are drawn in Section 7 and new directions of research are indicated.

2. High Level Formulation

In this section, we incorporate the uncertainty model involving $\mathcal{F}_r$ into a general robustness context. In Section 3, we particularize the formulation below to an LMI framework.

2.1 Deterministic Versus Probabilistic Margins: Given a control system with uncertain parameters $q \in Q_r$ and some specification denoted by $\mathcal{P}$, we can consider the

classical robustness margin

$$r_{\max} \doteq \sup\{r : \text{ Property } \mathcal{P} \text{ is satisfied for all } q \in Q_r\}.$$

For example, if $\mathcal{P}$ is a specification on overshoot to a step input, then $r_{\max}$ would represent the largest radius of uncertainty for which this specification is verified. In contrast, for each $0 \leq \varepsilon \leq 1$, we also define the *probabilistic robustness margin*

$$R_{\max}(\varepsilon) \doteq \sup\{r \geq 0 : \text{ Prob}\{\text{Property } \mathcal{P} \text{ holds}\} \geq 1 - \varepsilon \text{ for all } f \in \mathcal{F}_r\}.$$

The definition above is interpreted as follows: For any given acceptable risk level ε, the quantity $R_{\max}(\varepsilon)$ is the maximal radius of uncertainty for which the property $\mathcal{P}$ is guaranteed to hold with probability $p = 1 - \varepsilon$ for all $f \in \mathcal{F}_r$. In view of the definitions above, it is immediate that

$$R_{\max}(\varepsilon) \geq R_{\max}(0) \geq r_{\max}.$$

2.2 Sharp Probabilistic Estimators: Given a system with its performance specification $\mathcal{P}$, we note that use of the $\mathcal{F}_r$ framework for distributional robustness requires finding the worst-case density function. To this end, for a given uncertainty radius $r \geq 0$ and system specification $\mathcal{P}$, we can define the target set

$$T_r \doteq \{q \in Q_r : \mathcal{P} \text{ is satisfied}\}$$

and note that the theory in [3] and [4] applies for the case when T_r is a symmetric convex set. For such cases, the uniform distribution for q is the worst-case density associated with the desired radius $R_{\max}(\varepsilon)$. The focal point in this paper is the LMI problem; in the next section, we see that the target T_r turns out to be convex but not necessarily symmetric. We address the lack of symmetry by defining the notion of a "sharp probabilistic estimator" $\rho(\varepsilon)$: The quantity $\rho(\varepsilon)$ is seen to represent a risk-adjusted radius of uncertainty and such an estimator is required to satisfy two inequalities. The first inequality $\rho(\varepsilon) \leq R_{\max}(\varepsilon)$ is used to guarantee Property $\mathcal{P}$ with probability $p = 1 - \varepsilon$. The second inequality $\rho(\varepsilon) \geq r_{\max}$ is used to guarantee improvement over the deterministic margin. More precisely, we say that a function $\rho : [0, 1] \to [0, \infty)$ is a *sharp probabilistic estimator* for $R_{\max}$ if the inequality

$$R_{\max}(\varepsilon) \geq \rho(\varepsilon) \geq r_{\max}$$

is satisfied for all $0 \leq \varepsilon \leq 1$. Since $\rho(\varepsilon) = R_{\max}(\varepsilon)$ trivially satisfies the requirement above, it is important to construct $\rho(\varepsilon)$ in such a way that it is readily computable. This issue is addressed in Section 4.

2.3 Amplification: One of the main points of this paper is that sharp estimators enable us to take a solution $r_{\max}$ of the classical robustness problem and "amplify" it to reflect risk; recall that $\rho(\varepsilon) \geq r_{\max}$. To illustrate, suppose that for $\varepsilon = 0.001$, a non-critical performance specification is analyzed and we obtain $\rho(\varepsilon) = 2r_{\max}$. Then this tells us that the radius of uncertainty can be doubled without assuming a "significant" risk. In such a case, we would say that an amplification factor of two is guaranteed. A surprising feature of the theory to follow is the fact that large amplifications of the robustness radius are often possible while simultaneously keeping the risk ε at a low level; see example in Section 6.

More formally, for $\varepsilon \geq 0$, the *probabilistic amplification* is defined as

$$A(\varepsilon) \doteq \frac{R_{\max}(\varepsilon)}{r_{\max}}.$$

In the context of an estimator $\rho(\varepsilon)$ defined above, we call the quantity

$$A_\rho(\varepsilon) \doteq \frac{\rho(\varepsilon)}{r_{\max}}$$

a *sharp estimator* for $A(\varepsilon)$; i.e., $A_\rho(\varepsilon) \leq A(\varepsilon)$ and we note that sharpness of $\rho(\varepsilon)$ guarantees that $A_\rho(\varepsilon) \geq 1$ for all $0 \leq \varepsilon \leq 1$. That is, a sharp estimator enables one to enhance the deterministic robustness margin with a distribution-free well-defined level of risk.

3. Formulation Particularized to LMI Framework

To formalize the ideas above in an LMI context, we consider a parameterized family of real symmetric $N \times N$ matrices described by

$$M(q) \doteq M_0 + \Delta M(q),$$

where

$$\Delta M(q) \doteq q_1 M_1 + q_2 M_2 + \cdots + q_\ell M_\ell$$

with each matrix M_i assumed to be symmetric. Furthermore, the "nominal" matrix M_0 is assumed to be negative-definite and the uncertain parameter vector $q \doteq (q_1, q_2, \ldots, q_\ell)$ is restricted to the box Q_r of variable radius $r > 0$ as indicated in Section 1.1. To interface with the theory in Section 2.1, we say that the family of matrices

$$\mathcal{M}_r \doteq \{M(q) : \ q \in Q_r\}$$

has Property $\mathcal{P}$ if the condition $M(q) < 0$ is satisfied. Hence, this leads to classical robustness margin

$$r_{\max} = \sup\{r : \ M(q) < 0 \text{ for all } q \in Q_r\}$$

and probablilistic robustness margin

$$R_{\max}(\varepsilon) \doteq \sup\{r : \ \text{Prob}\{M(q_f) < 0\} \geq 1 - \varepsilon \text{ for all } f \in \mathcal{F}_r\}.$$

We now argue that many LMI problems with real parametric uncertainty fit into this framework. To this end, we note that in some cases, $M(q)$ above might represent some block partitioned matrix which depends on the solution of a set of linear matrix inequalities. For this reason, the dimension N of $M(q)$ might differ from the state dimension n of the underlying physical system under consideration.

3.1 Example 1 (Quadratic Stability): We consider an uncertain linear time-invariant system specified in the form of the state-space equation

$$\dot{x} = A(q)x,$$

where

$$A(q) \doteq A_0 + \Delta A(q)$$

is an $n \times n$ uncertain state matrix with stable nominal A_0,

$$\Delta A(q) \doteq \sum_{i=1}^{\ell} q_i A_i$$

and each A_i is a fixed $n \times n$ matrix. In the traditional Lyapunov setting, the problem of checking robustness of quadratic stability leads to the classical margin

$$r_{\max} = \sup\{r : \ A(q)S + SA^T(q) < 0 \text{ for some } S = S^T > 0 \text{ and all } q \in Q_r\}.$$

To compute $r_{\max}$, we let $q^k(r)$ denote the k-th extreme point of the box Q_r. Then, with $A_k \doteq A(q^k(r))$, the radius $r_{\max}$ is obtained as the optimal value of a convex optimization problem; i.e., find $\sup\{r : \ A_k S + SA_k^T < 0 \text{ for all } k; \quad S = S^T > 0\}$. This is a standard LMI problem and we note that any feasible pair (r, S) above leads to a guarantee of stability with uncertain bound Q_r and Lyapunov function $V(x) = x^T P x$ with $P = S^{-1}$. In view of this fact, we can apply the theory in this paper by using a numerical LMI solver to obtain an approximate optimum $r \approx r_{\max}$ with associated solution matrix $S = S^T > 0$. Then take

$$M(q) \doteq A(q)S + SA^T(q)$$

as a starting point for the theory to follow. Note that (r, S) does not necessarily need to be optimal in order to apply the theory in this paper; see discussion in Section 7.

3.2 Example 2 (H_∞ Performance): Consider the uncertain system described by

$$\begin{aligned} \dot{x} &= A(q)x + B(q)w; \\ y &= C(q)x \end{aligned}$$

with external input $w(t) \in \mathbf{R}^m$, output $y(t) \in \mathbf{R}^p$, state $x(t) \in \mathbf{R}^n$ and initial condition $x(0) = 0$. Furthermore, as in Example 3.1, the uncertain matrices above depend affinely on the vector of uncertain parameters $q \in \mathbf{R}^\ell$; i.e., we take

$$\begin{aligned} A(q) &\doteq A_0 + \Delta A(q) \doteq A_0 + \textstyle\sum_{i=1}^{\ell} q_i A_i; \\ B(q) &\doteq B_0 + \Delta B(q) \doteq B_0 + \textstyle\sum_{i=1}^{\ell} q_i B_i; \\ C(q) &\doteq C_0 + \Delta C(q) \doteq C_0 + \textstyle\sum_{i=1}^{\ell} q_i C_i \end{aligned}$$

with A_0, B_0 and C_0 denoting the nominal system and matrices A_i, B_i and C_i which are fixed; note that a subset of the A_i, B_i and C_i can be set to zero to handle uncertainties in one of the state space matrices which does not appear in another. Finally, it is also assumed that A_0 is stable. For a given uncertainty bound Q_r, LMI theory can be used to find the robust quadratic H_∞ gain

$$g_r = \sup_{w,q} \frac{\|y\|_2}{\|w\|_2},$$

where the supremum above is taken over all bounded inputs $w(t)$ and uncertainties $q \in Q_r$.

In order to recast the problem above in the new probabilistic framework of this paper, we take $\overline{g} > 0$ to be some acceptable level of gain. Typically, $\overline{g} > \|C_0(sI - A_0)^{-1}B_0\|_\infty \doteq g_0$ and we first seek the classical robustness margin $r_{\max}$ which can be obtained via solution of an LMI problem. Namely, we seek the maximum radius $r > 0$ subject to

$$\begin{bmatrix} A_k S + S A_k^T & S C_k^T & B_k \\ C_k S & -\overline{g} I & 0 \\ B_k^T & 0 & -\overline{g} I \end{bmatrix} < 0$$

for each extreme triple (A_k, B_k, C_k) obtained with the extreme point $q = q^k(r)$ and variable matrix $S = S^T > 0$. Note that stability is subsumed by this formulation.

Then, using a numerical LMI solver, we obtain an approximate optimum $r \approx r_{\max}$, associated solution matrix $S = S^T > 0$ and define

$$M_0 \doteq \begin{bmatrix} A_0 S + S A_0^T & S C_0^T & B_0 \\ C_0 S & -\overline{g} I & 0 \\ B_0^T & 0 & -\overline{g} I \end{bmatrix}$$

and

$$\Delta M(q) \doteq \begin{bmatrix} \Delta A(q) S + S \Delta A^T(q) & S \Delta C^T(q) & \Delta B(q) \\ \Delta C(q) S & 0 & 0 \\ \Delta B^T(q) & 0 & 0 \end{bmatrix}.$$

This leads to the matrix $M(q)$ with dimension $N = 3n$ and we define $R_{\max}(\varepsilon)$ as in Section 2.1 by declaring that Property $\mathcal{P}$ is satisfied if and only if $M(q) < 0$.

4. Main Result: A Sharp Estimator and Its Computability

In the first part of this section, we provide description of a function $\rho(\varepsilon)$ which is proven to be a sharp estimator for the general formulation of the probabilistic robustness problem involving $M(q)$. The second part of this section is relegated to describing a simple technique for numerical computation and graphical display of the $\rho(\varepsilon)$ plot and the associated sharp estimator $A_\rho(\varepsilon)$ for amplification.

4.1 Description of $\rho(\varepsilon)$: Recalling the notation u_r for the uniform distribution, we first define the random variable

$$\Phi(q_{u_1}) \doteq \|(-M_0^{-1/2})\Delta M(q_{u_1})(-M_0^{-1/2})\|_2$$

and denote its cumulative distribution function by $F_\Phi(x)$; notice that q is endowed with uniform distribution above. We are now prepared to state the main result.

4.2 Theorem (see Section 5 for proof): *The function*

$$\rho(\varepsilon) = \frac{1}{F_\Phi^{-1}(1-\varepsilon)}$$

is a sharp probabilistic estimator for $R_{\max}(\varepsilon)$.

4.3 Computability of $\rho(\varepsilon)$: It is now quite simple to obtain numerically accurate estimates of $\rho(\varepsilon)$ via sampling. In this regard, the important point to note is that the number of samples required is quite modest — even when the dimension of q is high; see [7]. Indeed, to obtain $\rho(\varepsilon)$, it suffices to numerically construct a cumulative distribution function for the random variable Φ. This is easily accomplished via the following sequence of steps:

Step 1. Take K samples of q_{u_1} uniformly distributed over the unit box Q_1.

Step 2. For each $x \geq 0$, let K_x be the number of samples for which

$$\Phi(q_{u_1}) \leq x.$$

Step 3. Construct the cumulative distribution function of Φ by setting

$$F_\Phi(x) \doteq \frac{K_x}{K}.$$

Step 4. For each $0 \leq y \leq 1$, let

$$F_\Phi^{-1}(y) \doteq \max\{x : \ F_\Phi(x) \leq y\}.$$

Step 5. For each $0 \leq \varepsilon \leq 1$, set

$$\rho(\varepsilon) \doteq \frac{1}{F_\Phi^{-1}(1-\varepsilon)}.$$

4.4 The Sharp Estimator $A_\rho(\varepsilon)$ for Amplification: In order to display $A_\rho(\varepsilon)$, we note that the classical robustness margin $r_{\max}$ is required in addition to $\rho(\varepsilon)$ given above. A formula for $r_{\max}$ is readily derived via minor modifications of the ideas in the proof of Theorem 4.2; i.e., taking into account the fact that a convex function on a box is maximized at one of its extreme points and that the box is symmetric, we readily obtain

$$r_{\max} = \frac{1}{\max\limits_k \|(-M_0^{-1/2})\Delta M(q^k)(-M_0^{-1/2})\|_2},$$

where q^k is the k-th extreme of the unit box Q_1.

5. Proof of Theorem 4.2

In order to prove the theorem, the lemma to follow plays an important role.

5.1 Lemma: *With $M(q)$ defined in Section 3 and any $q \in \mathbf{R}^\ell$, the pair of conditions $M(q) < 0$ and $M(-q) < 0$ are satisfied if and only if*

$$\|(-M_0^{-1/2})\Delta M(q)(-M_0^{-1/2})\|_2 < 1.$$

Proof: First observe that negative-definiteness of M_0 guarantees well-define matrix square roots. Now, the satisfaction of the pair of conditions $M(q) < 0$ and $M(-q) < 0$ is equivalent to the satisfaction of

$$\begin{aligned} x^T M_0 x + x^T \Delta M(q) x &< 0; \\ x^T M_0 x - x^T \Delta M(q) x &< 0 \end{aligned}$$

for all nonzero $x \in \mathbf{R}^N$. These conditions are in turn equivalent to

$$\max_{x \neq 0} \frac{|x^T \Delta M(q) x|}{x^T(-M_0)x} < 1.$$

Now, noting the nonsingularity of $(-M_0)^{1/2}$ and taking $y \doteq (-M_0)^{1/2}x$, it follows that the condition above holds if and only if

$$\max_{y \neq 0} \frac{|y^T(-M_0^{-1/2})\Delta M(q)(-M_0^{-1/2})y|}{y^T y} < 1.$$

In view of the inequality above, the pair of conditions $M(q) < 0$ and $M(-q) < 0$ are satisfied if and only if

$$\|(-M_0^{-1/2})\Delta M(q)(-M_0^{-1/2})\|_2 < 1.$$

5.2 Proof of Theorem 4.2: Given any $f \in \mathcal{F}_r$, we begin with the obvious inequality

$$\text{Prob}\{M(q_f) < 0\} \geq \text{Prob}\{M(q_f) < 0, M(-q_f) < 0\}$$

and let u_r be the uniform distribution over Q_r. Now, using the Uniformity Principle (see [3]) for the symmetric convex constraints above, we have

$$\begin{aligned} R_{\max}(\varepsilon) &= \sup\{r : \text{ Prob}\{M(q_f) < 0\} \geq 1 - \varepsilon \text{ for all } f \in \mathcal{F}_r\} \\ &\geq \sup\{r : \text{ Prob}\{M(q_f) < 0,\ M(-q_f) < 0\} \geq 1 - \varepsilon \text{ for all } f \in \mathcal{F}_r\} \\ &= \sup\{r : \text{ Prob}\{M(q_{u_r}) < 0,\ M(-q_{u_r}) < 0\} \geq 1 - \varepsilon\}. \end{aligned}$$

Also, noting that $M(q)$ is a positively homogeneous function of q, we let u_1 denote the uniform distribution over the unit box and apply Lemma 5.1, to obtain

$$\begin{aligned} &\sup\{r : \text{ Prob}\{M(q_{u_r}) < 0,\ M(-q_{u_r}) < 0\} \geq 1 - \varepsilon\} \\ &\qquad = \sup\{r : \text{ Prob}\{r\|(-M_0^{-1/2})\Delta M(q_{u_1})(-M_0^{-1/2})\|_2 < 1\} \geq 1 - \varepsilon\} \\ &\qquad = \frac{1}{F_\Phi^{-1}(1-\varepsilon)} \\ &\qquad = \rho(\varepsilon) \end{aligned}$$

Furthermore, since $\rho(0) \geq r_{\max}$ is trivially verified, we use the fact that $\rho(\varepsilon)$ is a nonincreasing function of ε to establish that

$$R_{\max}(\varepsilon) \geq \rho(\varepsilon) \geq \rho(0) \geq r_{\max}.$$

Hence, it follows that $\rho(\varepsilon)$ is a sharp estimator for $R_{\max}(\varepsilon)$.

6. Numerical Examples

In this section, we provide a numerical example illustrating the use of Theorem 4.2, which involves H_∞ performance of uncertain systems.

6.1 H_∞ Performance: We now analyze a system which fits the formulation of Section 3.2. In Figure 1, we depict a mechanical system consisting of four blocks with uncertain dampers and springs. We consider 8 uncertain parameters with ranges $1 \leq b_i \leq 2$ and $1 \leq k_i \leq 2$ for $i = 1, \ldots, 4$ and all unit masses $m_i = 1$.

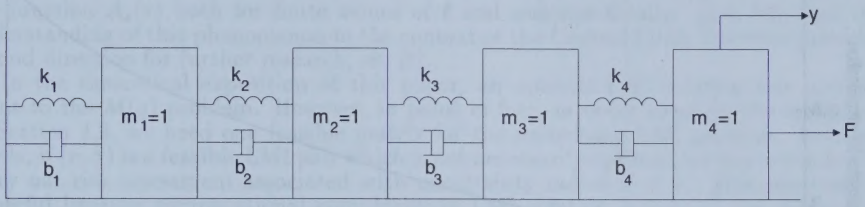


Figure 1. Mechanical System Used in Example 6.1.

Using the midpoint of the ranges to define the nominal system matrices, we identify uncertain parameters q_i with the b_i and k_i. Writing the equation of motion, the resulting system matrices are

$$A_0 = A(0) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1.5 & -1.5 & 1.5 & 1.5 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 1.5 & 1.5 & -3 & -3 & 1.5 & 1.5 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1.5 & 1.5 & -3 & -3 & 1.5 & 1.5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1.5 & 1.5 & -3 & -3 \end{bmatrix},$$

$$B_0 = B(0) = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}^T; \quad C_0 = C(0) = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix};$$

with $\Delta A(q)$ having the appropriate uncertainties corresponding to the non-zero entries of A_0 and $\Delta B(q) = \Delta C(q)^T = 0$. The first step of our numerical experiment involves computing the classical H_∞ gain for the nominal system described by A_0, B_0 and C_0; we obtained $g_0 \approx 5.912$.

Now, to study H_∞ performance in a probabilistic context, we consider a specification allowing for H_∞ variations which exceeds g_0 by 5% at most. Hence, in accordance with Section 3.2, we take $\overline{g} = 6.2076$. For this gain specification, we use the LMI Toolbox to obtain approximate solution matrix

$$S = \begin{bmatrix} 6.327 & -6.353 & -2.304 & 3.124 & -3.360 & 4.109 & -2.085 & 2.429 \\ -6.353 & 16.644 & 3.980 & -10.520 & 2.802 & -9.073 & 0.992 & -3.870 \\ -2.304 & 3.980 & 9.168 & -7.089 & -2.471 & 2.001 & -3.533 & 2.369 \\ 3.124 & -10.520 & -7.089 & 18.475 & 2.571 & -4.893 & 2.935 & -6.192 \\ -3.360 & 2.802 & -2.471 & 2.571 & 11.137 & -8.065 & 2.437 & -0.263 \\ 4.109 & -9.073 & 2.001 & -4.893 & -8.065 & 21.485 & -0.166 & -0.160 \\ -2.085 & 0.992 & -3.533 & 2.935 & 2.437 & -0.166 & 14.587 & -10.670 \\ 2.429 & -3.870 & 2.369 & -6.192 & -0.263 & -0.160 & -10.670 & 28.359 \end{bmatrix}$$

and classical robustness margin $r_{\max} \approx 0.0429$. Next, we implement Steps 1–5 of Section 4.3. Using 20,000 samples for q_{u_1}, we generate a plot $A_\rho(\varepsilon)$ with separation $\Delta\varepsilon = 0.001$ between ε points; see Figures 2 and 3. As a typical illustration of the power of the probabilistic method, observe that if one is willing to accept a 10% risk ($\varepsilon = 0.1$), the amplification of $r_{\max}$ is guaranteed to be at least 65%; i.e., $A_\rho(\varepsilon) \approx 1.65$.

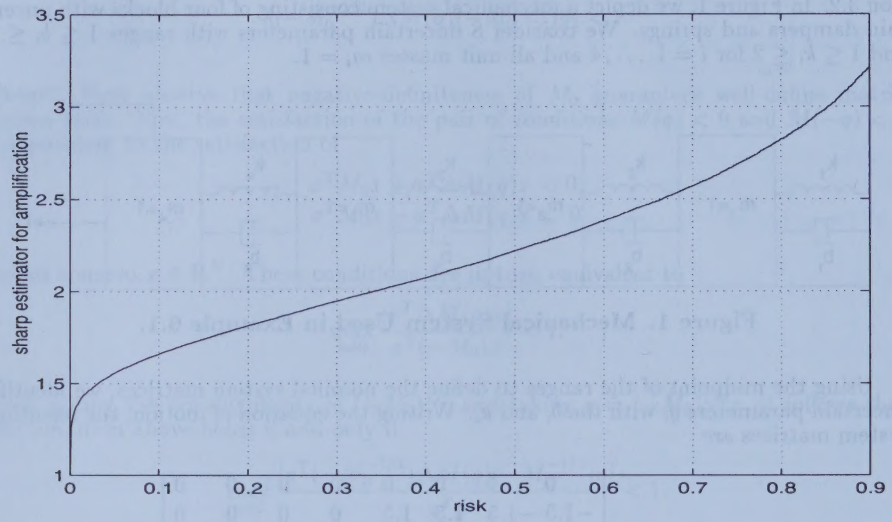


Figure 2. Sharp Estimator $A_\rho(\varepsilon)$ Versus Risk ε for Example 6.1 with $0 \leq \varepsilon \leq 0.9$.

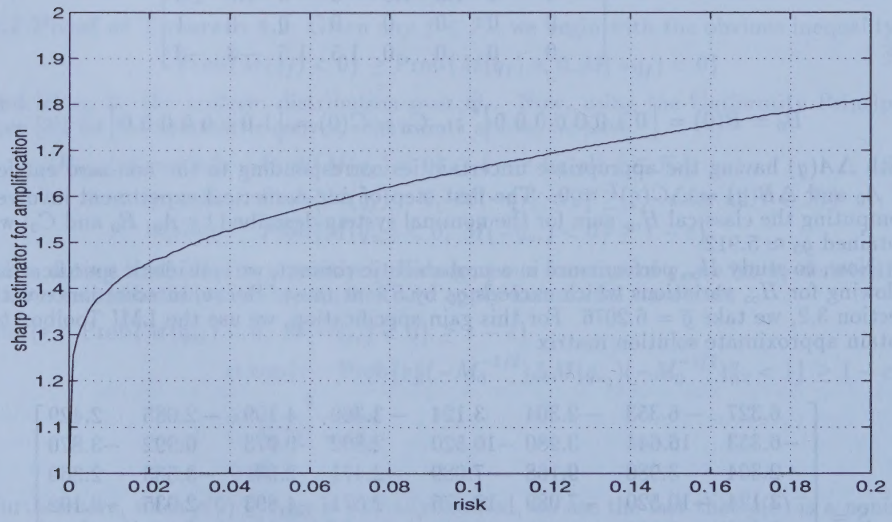


Figure 3. Sharp Estimator $A_\rho(\varepsilon)$ Versus Risk ε for Example 6.1 with $0 \leq \varepsilon \leq 0.2$.

7. Conclusion

The main objective in this paper was to demonstrate the efficacy of a new paradigm for probabilistic robustness analysis. To this end, LMI theory was considered and it was shown that distributionally robust margins of performance are easy to compute. A surprising feature of numerical experiments is that it is often possible to obtain large amplifications of the deterministic margin $r_{\max}$ while keeping the specified risk level ε quite small. Heuristically, it is clear that the greater the number ℓ of uncertainties, the higher the amplification. In this respect, it seems important to study the behavior of the function $A_\rho(\varepsilon)$ both for finite values of ℓ and assymptotically. It is felt that the understanding of this phenomenon in the context of the Central Limit Theorem provides a good direction for further research; see [8].

In the theoretical exposition of this paper, an optimal LMI solution was used as input to the $M(q)$ problem. However, in point of fact, in order to apply the technique in Section 4.3, we need one feasible matrix for the underlying LMI problem. In other words, if (r, S) is a feasible LMI pair which is not necessarily optimal, we can nevertheless carry out risk assessment associated with uncertainty radius $r' \geq r$. This observation is useful because computational complexity of LMI solution is high in the context of real parametric uncertainty. Hence, a good direction for future work involves use of suboptimal LMI solutions. For example, by working with a nominal system (set $q = 0$), one avoids the combinatorics of extreme points and in many cases, probabilistic estimates can be obtained which nevertheless amplify the deterministic margin $r_{\max}$.

A second topic of interest involves the quality of the estimator $\rho(\varepsilon)$. Given any sharp estimator $\rho(\varepsilon)$ for $R_{\max}(\varepsilon)$, an improvement may be possible in the following sense: A second estimator $\tilde{\rho}(\varepsilon)$ may be found leading to $\tilde{\rho}(\varepsilon) \geq \rho(\varepsilon)$ for some $0 \leq \varepsilon \leq 1$.

References

[1] Boyd, S. P., L. El Ghaoui, E. Feron and V. Balakrishnan (1994). *Linear Matrix Inequalities in System and Control Theory*, SIAM, Philadelphia.

[2] Gahinet, P. and A. Nemirovskii (1993). *LMI-Lab: A Package for User-Friendly LMI Design*, MatLab Reference Manual, The Math Works Inc., Natick.

[3] Barmish, B. R. and C. M. Lagoa (1995). "The Uniform Distribution: A Rigorous Justification for its Use in Robustness Analysis." *Technical Report ECE-95-12*, Department of Electrical and Computer Engineering, University of Wisconsin-Madison.

[4] Barmish, B. R. and C. M. Lagoa (1996). "The Uniformity Principle: A New Tool for Probabilistic Robustness Analysis." *30th Annual Conference on Information Science and Systems*, Princeton, 1996.

[5] Stengel, R. F. and L. R. Ray (1991). "Stochastic Robustness of Linear Time-Invariant Systems," *IEEE Transactions on Automatic Control*, AC-36, pp. 82-87.

[6] Ray, L. R. and R. F. Stengel (1993). "A Monte Carlo Approach to the Analysis of Control Systems Robustness," *Automatica*, vol. 3, pp. 229-236.

[7] Tempo, R. and E. W. Bai (1995). "Robustness Analysis with Nonlinear Parametric Uncertainty: A Probabilistic Approach," *CENS-CNR Report*, Politecnico di Torino.

[8] Barmish, B. R. and B. T. Polyak (1996). "A New Approach to Open Robustness Problems Based on Probabilistic Prediction Formulae," to appear in *Proceedings of IFAC World Congress*, San Francisco.

Further Results for Multi-Rate Decorrelators for Synchronous DS/CDMA Systems*

Jiangxin Chen & Urbashi Mitra
Department of Electrical Engineering
The Ohio State University, Columbus, OH 43210

Abstract

Recently, decorrelator-based receivers have been investigated for demodulating data in a dual-rate synchronous DS/CDMA system. The proposed receivers were a high-rate decorrelator (HRD) matched to the data rate of the high-rate users and a low-rate decorrelator (LRD) matched to the data rate of the low-rate users. In this paper, the probabilities of error attainable for high-rate and low-rate users with the use of the HRD and the LRD are analyzed. It is proven that the LRD offers superior performance to that of the HRD for all users. Asymptotic analysis provides bounds for the performance difference for high-rate users. The results are generalized to multi-rate systems. To increase fidelity, a Sliding-Window Decorrelator is proposed which demodulates a high-rate user's data by a soft decoding rule from the outputs of several decorrelators sliding along the received signal sequence. The results show that it performs better than the HRD while maintaining smaller demodulation delay and computational complexity than the LRD.

1 Introduction

Code-Division Multiple-Access (CDMA) is a promising technique for providing multi-user access in wireless systems. Relative to other access schemes, CDMA can increase the number of potential users in bursty or fading channels. Thus, CDMA is attractive for applications such as mobile cellular telephony and personal communications. Past forays into multi-user receiver development for CDMA-based communications (see *e.g.* [2, 4, 6, 8]) have focused on systems where each user transmits information at the same data rate. With the pursuit of an ubiquitous wireless communications system which can provide wireless transport for a variety of information sources, it will be desirable to develop multiple data rate systems. It is clear that voice, video and data have inherently different data rates.

In this paper, we shall investigate receiver structures for multi-rate Direct-Sequence CDMA (DS/CDMA) systems. In particular, we shall extend previous results for dual-rate systems [7]. This prior work considered decorrelator-based [4] receivers. The decorrelating detector [4] benefits from a simple linear structure, construction without knowledge of the received signal amplitudes and optimal near-far resistance [5].

The prior dual-rate work [7] proposed the use of two decorrelators: a high-rate decorrelator (HRD) matched to the data rate of the high-rate users and a low-rate decorrelator (LRD) matched to the data rate of the low-rate users. Herein, we shall prove that the performance attained for both high-rate and low-rate users employing LRDs is superior to that achievable with the use of HRDs. These results are generalized to multi-rate

*This research was supported by the National Science Foundation under Grant NCR 95-05881.

systems. In addition, asymptotic results provide simple bounds on the performance improvement for varying data rate ratio. While the LRD proffers improved performance for high-rate users, it does incur additional computational complexity as well as a processing delay. To improve performance without the attendant complexity and delay, we consider an improved linear receiver based on the HRD/LRD receivers.

2 Preliminaries

In a synchronous dual-rate DS/CDMA system, the bit interval of the low-rate users is denoted as T_0. The bit interval of the high-rate users is denoted as T_1, where we assume that $M = \frac{T_0}{T_1}$ is an integer. The received signal can be defined as

$$r(t) = \sum_{j=1}^{K_0} A_{j,0} b_{j,0}(t) s_{j,0}(t) + \sum_{k=1}^{K_1} \left\{ \sum_{m=1}^{M} A_{k,1}^m b_{k,1}^m(t) s_{k,1}^m(t) \right\} + n(t), \quad (1)$$

where the second subscript indicates whether the parameter is for a low-rate user (0) or for a high-rate user (1). $A_{j,0}$ and $b_{j,0}(t)$ represent the received amplitude for the j'th low-rate user and its received bit at time t. Similarly, $A_{k,1}^m$ and $b_{k,1}^m(t)$ are the received amplitude in the m'th sub-interval for the k'th high-rate user and its received bit at time t. The signature sequences are denoted by $s_{j,0}(t)$ and $s_{k,1}^m(t)$ for low-rate and high-rate users, respectively. $n(t)$ is additive white Gaussian noise with variance σ^2. Each user's signature sequence is normalized over its bit interval, *i.e.*

$$\begin{aligned} \int_0^{T_0} \left(s_{k,0}(t)\right)^2 dt &= 1 \quad k \in \{1, \cdots, K_0\} \\ \int_0^{T_1} \left(s_{k,1}^m(t)\right)^2 dt &= 1 \quad k \in \{1, \cdots, K_1\}, m \in \{1, \cdots, M\}. \end{aligned} \quad (2)$$

This dual-rate system model admits exploration from two perspectives. First note that in the interval $[0, T_0]$, each high-rate user is equivalent to M virtual low-rate users. The signature sequences for the m'th virtual user (m'th bit) of high rate user k is

$$\tilde{s}_{k,1}^m(t) = \begin{cases} s_{k,1}^m(t) & \text{for } (m-1)T_1 \le t < mT_1 \\ 0 & \text{else.} \end{cases} \quad (3)$$

Therefore, the dual-rate system with K_0 low-rate users and K_1 high-rate users is equivalent to a single-rate system with K users ($K = K_0 + MK_1$). This single, common rate is equivalent to the data rate of the low-rate user. After the received signal passes through the bank of filters matched to the spreading codes of the active users, the outputs can be written in the following matrix form,

$$\underline{r} = \Gamma \Lambda \underline{b} + \underline{n}, \quad (4)$$

where $\underline{b} = [b_{1,0}, \cdots, b_{K_0,0}, b_{1,1}^1, \cdots, b_{K_1,1}^1, b_{1,1}^2 \cdots, b_{K_1,1}^2, \cdots, b_{1,1}^M, \cdots, b_{K_1,1}^M]^T$, Γ is the spreading code correlation matrix, Λ is a diagonal matrix whose diagonal elements are the received amplitudes of the active users, and $\underline{n}$ is a Gaussian noise vector with zero mean and correlation matrix $\sigma^2\Gamma$. We can perform joint detection of $\underline{b}$ by applying a decorrelator to this virtual single-rate system:

$$\hat{\underline{b}} = \operatorname{sgn}\left(\Gamma^{-1}\underline{r}\right) = \operatorname{sgn}\left(\Lambda\underline{b} + \Gamma^{-1}\underline{n}\right). \quad (5)$$

This type of decorrelator will be termed a *low-rate decorrelator*(LRD)[7].

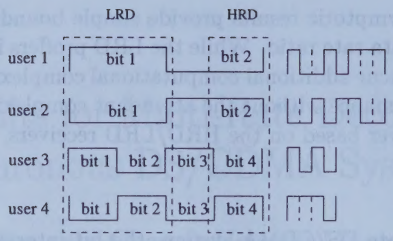


Figure 1: Abstract view of HRD and LRD operation.

Similarly, we can also create a decorrelator matched to each sub-interval $[(i-1)\,T_1, i\,T_1]$. In the i'th sub-interval, each of the K_1 high-rate users transmits one bit, $b^i_{k,1}$, while each of the K_0 low-rate users transmits a portion of the signal due to bit, $b_{j,0}$. This concept is depicted in Figure 1. After the received signal passes through the bank of matched filters, the outputs in this sub-interval can be written in matrix form as,

$$\underline{r}^i = \Gamma^i \Lambda^i \underline{b}^i + \underline{n}^i, \tag{6}$$

where Γ^i is the $(K_0 + K_1) \times (K_0 + K_1)$ correlation matrix for the i'th sub-interval, $\underline{n}^i$ is a Gaussian noise vector with zero mean and correlation matrix $\sigma^2\Gamma^i$. Note that in this sub-interval, the signature sequence for a high-rate user has unit energy, while the signature sequence for each low-rate user has only a fraction $(\frac{1}{M})$ of its total unit energy. The users in this sub-interval can be regarded as $K_0 + K_1$ high-rate users and thus can be jointly demodulated by a decorrelator matched to the high-rate:

$$\hat{\underline{b}}^i = \operatorname{sgn}\left(\Gamma^{i^{-1}} \underline{r}^i\right) = \operatorname{sgn}\left(\Lambda^i \underline{b}^i + \Gamma^{i^{-1}} \underline{n}^i\right). \tag{7}$$

This decorrelator is referred to as a *high-rate decorrelator*(HRD)[7]. In the next few sections, we will investigate the performance of the LRD *vis à vis* the HRD.

3 Review of Previous Work on Dual-Rate Decorrelators

In [7] the use of the LRD and HRD as described above was proposed. It was observed that the use of the LRD incurred a demodulation delay for the high-rate users; in addition, it required the inversion of a potentially large matrix. It is patent that as the ratio M increases, the dimension of the correlation matrix Γ increases rapidly even if K_1 and K_0 are small. Increasing M further exacerbates the complexity and delay concerns associated with the LRD. This property of the LRD recommends the use of the HRD. For the high-rate users, the received bits are the output of the decorrelator applied in each sub-interval. For the low-rate users, a soft decoding rule based on *maximal ratio combining* [1] is used to estimate the low-rate user bits:

$$\hat{b}_j = \operatorname{sgn}\left[\sum_{i=1}^{M} \left(\frac{1}{\left[\Gamma^{i^{-1}}\right]_{j,j}}\right) \left[\Gamma^{i^{-1}} \underline{r}^i\right]_j\right] \quad 1 \le j \le K_0. \tag{8}$$

Saquib *et. al.*[7] proved that if the signature sequence for the low-rate users are the same in every sub-interval, then the LRD and HRD achieve the same bit error rate for each low-rate user. This type of signature code for the low-rate user will be referred to as a *repetition code*. The authors [7] observed from their numerical examples that the high-rate users suffer in performance when a HRD is applied instead of the LRD. We next prove and generalize this observation.

4 Further Results on LRD and HRD

In this section, we prove that the performance resulting from demodulating high-rate and low-rate users using the LRD is superior to that achieved by the HRD. This result are generalized to systems where multiple data rates are offered.

Prior to presenting the results, we summarize the notation used in [7] and the present work. Since the M virtual users of each high-rate user are orthogonal to each other, Γ (the correlation matrix at the low data rate) is a sparse matrix with the following block matrix form:

$$\Gamma = \begin{bmatrix} \Gamma_{00} & \Gamma_{01} & \Gamma_{02} & \cdots & \Gamma_{0M} \\ \Gamma_{10} & \Gamma_{11} & 0 & \cdots & 0 \\ \Gamma_{20} & 0 & \Gamma_{22} & \cdots & 0 \\ \vdots & \vdots & & \ddots & \vdots \\ \Gamma_{M0} & 0 & \cdots & \cdots & \Gamma_{MM} \end{bmatrix} \quad \begin{array}{rl} \text{where } [\Gamma_{00}]_{m,n} &= \int_0^{T_0} s_{m,0}(t) s_{n,0}(t) dt \\ [\Gamma_{0k}]_{m,n} &= \int_0^{T_0} s_{m,0}(t) \tilde{s}^k_{n,1}(t) dt \quad k > 0 \\ \text{and } [\Gamma_{kk}]_{m,n} &= \int_0^{T_0} \tilde{s}^k_{m,1}(t) \tilde{s}^k_{n,1}(t) dt \quad k > 0 \end{array} , \quad (9)$$

and $\Gamma_{ij}^T = \Gamma_{ji}$. Correspondingly, the inverse of Γ also admits a block form. The matrices B_{ij} are found to be [7]:

$$\Gamma^{-1} = \begin{bmatrix} B_{00} & \cdots & B_{0M} \\ \vdots & & \vdots \\ B_{M0} & \cdots & B_{MM} \end{bmatrix} \quad \begin{array}{rl} \text{where } B_{00} &= \left(\Gamma_{00} - \sum_{i=1}^{M} \Gamma_{0i} \Gamma_{ii}^{-1} \Gamma_{i0}\right)^{-1} \\ B_{0i} &= -B_{00} \Gamma_{0i} \Gamma_{ii}^{-1} \quad i > 0 \\ B_{ij} &= \Gamma_{ii}^{-1} \Gamma_{i0} B_{00} \Gamma_{0j} \Gamma_{jj}^{-1} \quad i, j > 0, i \neq j \\ \text{and } B_{ii} &= \Gamma_{ii}^{-1} + \Gamma_{ii}^{-1} \Gamma_{i0} B_{00} \Gamma_{0i} \Gamma_{ii}^{-1} \quad i > 0 \end{array} . \quad (10)$$

For the high-rate decorrelator, the correlation matrix Γ^i for the i'th sub-interval has the form:

$$\Gamma^i = \begin{bmatrix} \Gamma^i_{00} & \Gamma_{0i} \\ \Gamma_{i0} & \Gamma_{ii} \end{bmatrix}, \quad (11)$$

where Γ_{0i}, Γ_{i0} and Γ_{ii} are the same as those in Γ. But $\Gamma_{00} = \sum_{i=1}^{M} \Gamma^i_{00}$ because Γ^i_{00} is the correlation matrix for low-rate users only in i'th sub-interval. The inverse of Γ^i is given by,

$$[\Gamma^i]^{-1} = \begin{bmatrix} L^i & -L^i \Gamma_{0i} \Gamma_{ii}^{-1} \\ -H^i \Gamma_{i0} \left(\Gamma^i_{00}\right)^{-1} & H^i \end{bmatrix} \quad \begin{array}{rl} \text{where } L^i &= \left(\Gamma^i_{00} - \Gamma_{0i} \Gamma_{ii}^{-1} \Gamma_{i0}\right)^{-1} \\ \text{and } H^i &= \left(\Gamma_{ii} - \Gamma_{i0} {\Gamma^i_{00}}^{-1} \Gamma_{0i}\right)^{-1} \end{array} . \quad (12)$$

Exploiting the properties of positive definite matrices, we can prove the following claim:

Proposition 4.1 *In a general dual-rate system, if we denote $P^i_{k,1}(R_H)$ as the probability of error for detecting the i'th bit for high-rate user k using an HRD and similarly $P^i_{k,1}(R_L)$ as the probability of error for detecting the i'th bit for high-rate user k using the LRD, then*

$$P^i_{k,1}(R_H) \geq P^i_{k,1}(R_L). \quad (13)$$

Proof:

By equation (10), (12) and matrix inversion lemma (see *e.g.* [3]),

$$B_{00} = \left(\sum_{j=1}^{M} (L^j)^{-1}\right)^{-1} = \left((L^i)^{-1} + \sum_{\substack{1 \leq j \leq M \\ j \neq i}} (L^j)^{-1}\right)^{-1}$$

$$= L^i - L^i \left\{ \left[\sum_{\substack{1 \le j \le M \\ j \ne i}} (L^j)^{-1} \right]^{-1} + L^i \right\}^{-1} L^i \tag{14}$$

$$H^i = \left(\Gamma_{ii} - \Gamma_{i0}\Gamma_{00}^{i\ -1}\Gamma_{0i}\right)^{-1} = \Gamma_{ii}^{-1} + \Gamma_{ii}^{-1}\Gamma_{i0}(\Gamma_{00}^i - \Gamma_{0i}\Gamma_{ii}^{-1}\Gamma_{i0})^{-1}\Gamma_{0i}\Gamma_{ii}^{-1}$$

$$= \Gamma_{ii}^{-1} + \Gamma_{ii}^{-1}\Gamma_{i0}L^i\Gamma_{0i}\Gamma_{ii}^{-1} \tag{15}$$

$$B_{ii} - H^i = \Gamma_{ii}^{-1}\Gamma_{i0}(B_{00} - L^i)\Gamma_{0i}\Gamma_{ii}^{-1} = -\Gamma_{ii}^{-1}\Gamma_{i0}L^i \left((L^i)^{-1} + \sum_{\substack{1 \le j \le M \\ j \ne i}} (L^j)^{-1} \right)^{-1} L^i\Gamma_{0i}\Gamma_{ii}^{-1}$$

$$= -D_i^T L^i \left((L^i)^{-1} + \sum_{\substack{1 \le j \le M \\ j \ne i}} (L^j)^{-1} \right)^{-1} L^i D_i, \tag{16}$$

where $D_i = \Gamma_{0i}\Gamma_{ii}^{-1} = [\underline{d}_1^i, \underline{d}_2^i, \ldots, \underline{d}_{K_1}^i]$. Let us denote the following: $[B_{ii}]_{k,k} = b_{i,k}$, $[H^i]_{k,k} = h_{i,k}$, and note that the matrix $L^i \left((L^i)^{-1} + \sum_{\substack{1 \le j \le M \\ j \ne i}} (L^j)^{-1} \right)^{-1} L^i$ is positive definite, then

$$b_{i,k} - h_{i,k} = -\underline{d}_k^{i\ T} L^i \left((L^i)^{-1} + \sum_{\substack{1 \le j \le M \\ j \ne i}} (L^j)^{-1} \right)^{-1} L^i \underline{d}_k^i \le 0. \tag{17}$$

The probabilities of detection error for the HRD and LRD for the k'th high-rate user in the i'th sub-interval are given, respectively, by,

$$P_{k,1}^i(R_H) = Q\left(\frac{A_{k,1}^i}{\sigma\sqrt{h_{i,k}}}\right) \qquad P_{k,1}^i(R_L) = Q\left(\frac{A_{k,1}^i}{\sigma\sqrt{b_{i,k}}}\right), \tag{18}$$

where $Q(x)$ is the complementary cumulative distribution function.[1] By equation (17),

$$P_{k,1}^i(R_H) \ge P_{k,1}^i(R_L). \tag{19}$$

The claim is proven. □

Note that by equation (14) we can also compare the performance of LRD and HRD for low-rate users (without maximal ratio combining):

$$P_{k,0}^i(R_H) > P_{k,0}(R_L), \tag{20}$$

where $P_{k,0}^i(R_H)$ is the probability of error for detecting the bit in the i'th sub-interval for low-rate user k using an HRD and similarly $P_{k,0}(R_L)$ is the probability of error for detecting the bit for low-rate user k using the LRD.

If the low-rate users use repetition codes, then $\Gamma_{0i} = \Gamma_{01}$, $\Gamma_{ii} = \Gamma_{11}$ for $i > 0$. We can therefore simplify B_{00},

$$B_{00} = \left(\Gamma_{00} - M\,\Gamma_{01}\Gamma_{11}^{-1}\Gamma_{10}\right)^{-1}. \tag{21}$$

In this scenario, if the signature sequences for the high-rate and low-rate users are fixed at each sub-interval, as the ratio of the rates $M \to \infty$, $B_{ii} \to \Gamma_{ii}^{-1}$, H^i remains the same, therefore, $P_{k,1}^i(R_L)$ will decrease while $P_{k,1}^i(R_H)$ remains the same by equations (10), (12)

[1] $Q(x) = \int_x^\infty \frac{1}{\sqrt{2\pi}} \exp\left\{-\frac{v^2}{2}\right\} dv$.

and (21). As $M \to \infty$, the difference between the probabilities of error for the HRD and the LRD will increase and approach a non-zero constant. Thus we can determine bounds on the difference in performance. This asymptotic behavior will be explored in Section 6.

We can extend Proposition 4.1 to the scenario where multiple data rates are offered.

Proposition 4.2 *For a general multi-rate system, with decorrelators matched to data rates R_1 and R_2 if $R_2 > R_1$,*

$$\text{then} \quad P_{k,i}(R_2) \geq P_{k,i}(R_1). \tag{22}$$

The probabilities of error above are for any type of user; it is assumed that there is no maximal ratio combining and that simply a R-rate decorrelator is employed. We will later define such a decorrelator to be a medium-rate decorrelator in Section 6.

Proof:

For a general multi-rate system with decorrelators matched to data rates R_1 and R_2 ($R_2 > R_1$), we can group the users into two categories: those with data rates $< R_2$ which will be denoted as lower-rate users and those with data rates $\geq R_2$ which will be denoted as higher-rate users. Those lower-rate users whose data rates are not equal to R_1 can be regarded as virtual users with data rate R_1. Those higher-rate users whose data rates are greater than R_2 can be regarded as virtual users with data rate R_2. (Note that we only consider multi-rate system in which higher rates are mulitiples of the lower rates.) Now the case of a multi-rate system is reduced to a dual-rate system with data rates R_1 and R_2. Note that in the proof of Proposition 4.1, the only constraint on the correlation matrices is that they are positive definite. So Proposition 4.2 can be proved by applying Proposition 4.1 directly. □

Furthermore, for the general multi-rate system we can show that

Proposition 4.3 *Given the i'th sub-interval, and two data rates, $R_1 \neq R_2$,*

$$P^i_{k,1}(R_1) = P^i_{k,1}(R_2) \quad \forall \ k \ \textit{simultaneously}, \tag{23}$$

if and only if the set of signature sequences for the high rate users is orthogonal to the set of signature sequences for the low-rate users. $P^i_{k,1}(R)$ is the probability of detection error for detecting the i'th bit of high-rate user k with a decorrelator matched to rate R.

Proof: It is straightforward by equation (17) and Proposition 4.2. □

From the above propositions, we conclude that the high-rate users suffer performance degradation when using a decorrelator matched to any rate but the lowest rate. This is the price paid for shorter demodulation delay and less computational complexity. Only orthogonality between the user sets can ensure equivalent performance for the high-rate users.

Similar results hold for detecting the low-rate user bits using the HRD and maximal ratio combining:

Proposition 4.4 *For a general multi-rate system, with decorrelators matched to data rates R and R_L, where R_L is the data rate for low-rate user k, if $R > R_L$,*

$$\text{then} \quad P_{k,0}(R) \geq P_{k,0}(R_L), \tag{24}$$

where $P_{k,0}(R)$ is the probability of bit detection error for low-rate user k with maximal ratio combining on the R rate decorrelator, and $P_{k,0}(R_L)$ is the probability of detection error for the low-rate decorrelator for low-rate user k.

Proof:

Let us first consider a dual-rate system in which the low-rate users can use repetition codes or non-repetition codes. The probability of bit detection error for low-rate user k with maximal ratio combining on the HRD is given by:

$$P_{k,0}(R) = Q\left[\frac{A_{k,0}}{\sigma}\sqrt{\sum_{i=1}^{M}\left(\frac{1}{[L^i]_{k,k}}\right)}\right]. \tag{25}$$

The probability of bit detection error for the LRD for low-rate user k is given by:

$$P_{k,0}(R_L) = Q\left(\frac{A_{k,0}}{\sigma}\sqrt{\frac{1}{[B_{00}]_{k,k}}}\right) = Q\left(\frac{A_{k,0}}{\sigma}\sqrt{\frac{1}{\left[\left(\sum_{i=1}^{M}(L^i)^{-1}\right)^{-1}\right]_{k,k}}}\right). \tag{26}$$

Note that as L^i is positive definite, by Bergstrom's inequality (see *e.g.* [3], pp. 453) and induction, we can prove that

$$\sum_{i=1}^{M}\left(\frac{1}{[L^i]_{k,k}}\right) \leq \frac{1}{\left[\left(\sum_{i=1}^{M}(L^i)^{-1}\right)^{-1}\right]_{k,k}}. \tag{27}$$

Therefore, the claim is proved. □

Note that in the case that the low-rate users use repetition codes, $L^i = L^1$ for $i > 0$, (27) achieves equality [7]. To extend the result to the multi-rate case, we use the same method as in the proof of Proposition 4.2.

5 Sliding-Window Decorrelator

As has just been shown, the high-rate users suffer performance degradation when employing the HRD. The probability of error can be reduced by extending the soft-decoding idea proposed in [7] for the HRD and low-rate users to that for high-rate users. In order to employ such a scheme, multiple estimates of the high-rate users bits are formed. The estimates are drawn from decorrelators designed for C sub-intervals. The soft-decoding scheme is best illustrated in Figure 2. Since the window of length CT_1 "slides" over the received signal, we name this new receiver the *sliding-window decorrelator* (SWD) [2]. A linear combination of these estimates via maximizing the output SNR is proposed. We let

$$\underline{r}^{i,i+C-1} = \left[\underline{r}^{iT}, \underline{r}^{i+1T}, \cdots, \underline{r}^{i+C-1T}\right], \tag{28}$$

and denote the correlation matrix for the signal present in $\underline{r}^{i,i+C-1}$ as Γ_C^i. We form multiple estimates for high-rate user bit, $b_{k,1}^i$

$$\underline{\beta} = \begin{bmatrix} \left[\left(\Gamma_C^{i-C+1}\right)^{-1}\underline{r}^{i-C+1,i}\right]_{k+(C-1)K_1+K_0} \\ \left[\left(\Gamma_C^{i-C+2}\right)^{-1}\underline{r}^{i-C+2,i+1}\right]_{k+(C-2)K_1+K_0} \\ \vdots \\ \left[\left(\Gamma_C^{i-C+L}\right)^{-1}\underline{r}^{i-C+L,i+L-1}\right]_{k+K_0} \end{bmatrix}, \tag{29}$$

[2]It is noted that a Sliding-Window Decorrelator is proposed in [9] to improve the decorrelator's performance in an asynchronous DS/CDMA single-rate system.

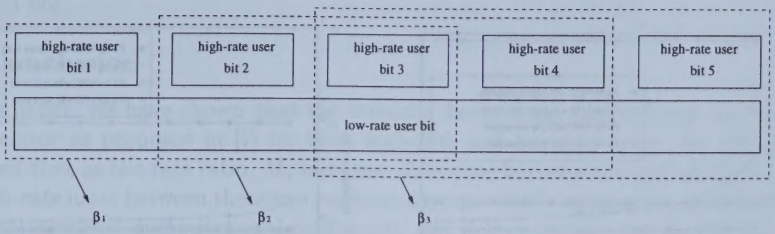


Figure 2: Soft-decoding scheme for high-rate users using a sliding window.

where $L = \min(C, M - C + 1)$. Defining L in this manner ensures that estimates for $b^i_{k,1}$ are formed from signals that fall within the low-rate bit interval. Thus if $C = M$, the SWD is equivalent to the LRD. Each component of $\underline{\beta}$ is an estimate for $b^i_{k,1}$. For simplicity let $A \equiv A^i_{k,1}$ and $b \equiv b^i_{k,1}$. Due to the decorrelation operations, the vector $\underline{\beta}$ has the following form:

$$\underline{\beta} = [Ab + \tilde{n}_1, Ab + \tilde{n}_2, \cdots, Ab + \tilde{n}_C]^T, \tag{30}$$

where $\underline{\tilde{n}}$ is a zero-mean Gaussian vector with covariance matrix $\Sigma = \sigma^2 \Psi \Psi^T$. Let,

$$\underline{m}_j = \left[\left(\Gamma_C^{i-C+1} \right)^{-1} \left(S^{i-C+1,i} \right)^T \right]_{k+(C-j)K_1+K_0} \tag{31}$$

$$\text{then } \Psi = \begin{bmatrix} \longleftarrow & \underline{m}_1 & \longrightarrow & \underline{0} & \cdots & \cdots & \underline{0} \\ \underline{0} & \longleftarrow & \underline{m}_2 & \longrightarrow & \underline{0} & \cdots & \underline{0} \\ \vdots & \ddots & \ddots & \ddots & & & \vdots \\ \underline{0} & \vdots & \ddots & \ddots & \ddots & & \underline{0} \\ \underline{0} & \cdots & \cdots & \underline{0} & \longleftarrow & \underline{m}_C & \longrightarrow \end{bmatrix}. \tag{32}$$

$S^{i-C+1,i}$ is a matrix whose columns contain the chip-matched filtered signature sequences of all the users signals present in the interval $[(i - C + 1)T_1, i\,T_1]$.

Once again, maximal ratio combining [1] is proposed for improving the detection error. The new estimate is formed via correlation: $\hat{b^*}^i_{k,1} = \text{sgn}(\underline{q}^T \underline{\beta})$. The desired coefficients and the resulting maximized output SNR are provided below:

$$\underline{q} = \frac{A}{\sigma^2} \left(\Psi \Psi^T \right)^{-1} \underline{\xi} \quad \Longrightarrow \quad SNR_{max} = \frac{A^2}{\sigma^2} \underline{\xi}^T \left(\Psi \Psi^T \right)^{-1} \underline{\xi}, \tag{33}$$

where $\underline{\xi}$ is a $C \times 1$ vector comprised of ones. For the case of repetition codes for low-rate users, the output SNR for the k'th high-rate user in the i'th sub-interval is given by:

$$SNR_{max} = \frac{A^2}{\sigma^2} \frac{1}{\left[\left(\Gamma_C^{i-C+1} \right)^{-1} \right]_{K_0+k,K_0+k} - \frac{L-1}{2}\alpha} \tag{34}$$

where $\alpha = \frac{1}{C} \left[\Gamma_{1,1}^{-1} \Gamma_{1,0} E_{0,0}^{i-C+1} \Gamma_{0,1} \Gamma_{1,1}^{-1} \right]_{k,k}$, and $E_{0,0}^{i-C+1}$ is the $K_0 \times K_0$ principal submatrix of $\left[\Gamma_C^{i-C+1} \right]^{-1}$. For the repetition code case, once $\underline{m}_1$ has been formed, the rest of the decorrelation filters $\underline{m}_i, i > 1$ are simply permutations of blocks of $\underline{m}_1$, thereby providing yet another implementation advantage resulting from the employment of repetition codes. By construction, the output SNR of the proposed estimator will be greater than that of a single estimate. This difference is investigated numerically in the sequel.

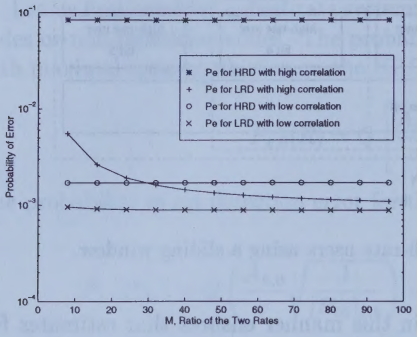

Pe for HRD with high correlation
Pe for LRD with high correlation
Pe for HRD with low correlation
Pe for LRD with low correlation
Probability of Error
M, Ratio of the Two Rates

Figure 3: Performance comparison for LRD and HRD when the signature sequences are highly correlated and minimally correlated.

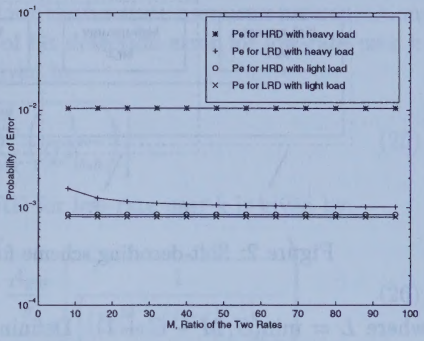

Pe for HRD with heavy load
Pe for LRD with heavy load
Pe for HRD with light load
Pe for LRD with light load
Probability of Error
M, Ratio of the Two Rates

Figure 4: Performance comparison for LRD and HRD under high load and low load.

6 Numerical Results

In this section, we present numerical results for the properties we have theoretically investigated in the prior sections. In Figures 3 and 4 we see the asymptotic effects of increasing the rate ratio M. In Figure 3 the effects of high and low cross-correlation between signature sequences is investigated. The SNR for the desired user is 10dB, while $K_0 = 3$ and $K_1 = 2$. The signature sequences are formed from length 7 Gold codes and m-sequences. For the high correlation case we see significant performance degradation for the HRD versus the LRD; however if the correlations are minimized, the degradation is not as significant. The effects of loading are explored in Figure 4. The high load scenario is parameterized by $K_0 = 8$ and $K_1 = 7$ while in the low load case there are 2 low-rate and 2 high-rate users. For this figure, the signature sequences were created from length 15 m-sequences. It is clear that the effects of loading are analogous to that of correlation between codes. In Figure 5 we observe the improvement afforded with the use of the Sliding-Window Decorrelator (SWD). The output SNR of the SWD is compared to that attained by a single estimate using a decorrelator matched to the interval CT_1, hence we call this decorrelator a *medium-rate decorrelator* (MRD). The SWD creates the final estimate by combining the outputs of L MRD estimates. Again performance is investigated by considering the load and the sequence cross-correlation. The parameters for the scenarios considered are the same as those in the previous figures. It is patent that for lightly loaded systems or systems where there is minimal cross-correlation between the codes, the SWD affords little advantage. However, for high loading or high cross-correlation, gains as high as 1dB can be experienced with short windows ($C = 2, 3$). Thus there are scenarios where it would be prudent to apply the SWD, and those where

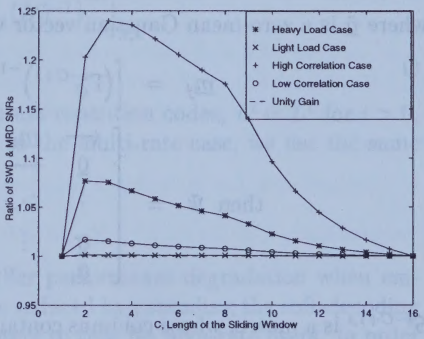


Figure 5: Ratio of output SNRs for SWD and MRD for varying loads and correlations between the signature sequences.

it would not.

7 Conclusion

In this paper, we have shown that the low-rate decorrelator outperforms the high-rate deccorelator as proposed in [7] for both high-rate and low-rate users. In addition, we observed that as the rate ratio, M, increases difference between the probabilities of error for high-rate users between these two receivers asymptotically approaches a constant. The inherent increased computational complexity and delays present due to the use of the LRD for high-rate user detection are combatted by introducing maximal ratio combining on multiple estimates of the high-rate user bits. As few as two or three pre-estimates can result in significant performance gains.

Future work will focus on the further analysis of SWD performance. We will also explore novel receivers for general multi-rate scenarios.

References

[1] D. G. Brennan, "Linear diversity combining techniques," *Proceedings of IRE*, vol. 47, pp. 1075–1102, June 1959.

[2] A. Duel-Hallen, "Decorrelating decision-feedback multiuser detector for synchronous Code-Division Multiple-Access Channels," *IEEE Trans. on Comm.*, vol. 41, no. 2, pp. 285–290, February 1993.

[3] R. A. Horn and C. R. Johnson, *Matrix Analysis.* Cambridge: Cambridge University Press, 1985.

[4] R. Lupas and S. Verdú, "Linear multiuser detectors for synchronous Code-Division Multiple-Access channels," *IEEE Trans. Inform. Theory*, vol. 35, no. 1, pp. 123–136, January 1989.

[5] R. Lupas and S. Verdú, "Near-far resistance of multi-user detectors in asynchronous channels," *IEEE Trans. on Comm.*, vol. 38, pp. 496–508, April 1990.

[6] U. Madhow and M. L. Honig, "MMSE interference suppression for direct-sequence spread spectrum CDMA," *IEEE Trans. on Comm.*, vol. 42, no. 12, pp. 3178–3188, December 1994.

[7] M. Saquib, R. Yates, and N. Mandayam, "Decorrelating detectors for a dual rate synchronous DS/CDMA channel," in *Proceedings of the IEEE Vehicular Technology Conference, Atlanta, GA*, May 1996.

[8] S. Verdú, "Minimum probability of error for asynchronous Gaussian multiple-access channels," *IEEE Trans. Inform. Theory*, vol. 32, no. 1, pp. 85–96, January 1986.

[9] S. S. H. Wijayasuriya, G. H. Norton, and J. P. McGeehan, "A sliding window decorrelating receiver for multiuser DS-CDMA mobile radio networks," *IEEE Trans. on Vehicular Technology*, vol. 45, no. 3, pp. 503–521, August 1996.

Blind Decorrelation of Asynchronous Multiuser Signals using Fractionally-Spaced Sampling

Maïté Brandt-Pearce
Department of Electrical Engineering
University of Virginia
Charlottesville, VA 22903

ABSTRACT

A decorrelating filter is described that separates signals in a multiuser environment as long as the relative delays are sufficiently distinct. The input signal is fractionally sampled to allow for the differentiation of the user delays. It is shown that in a noisy environment, the signal-to-noise ratio penalty for large symbol frame size is severe. The performance of a code-division multiple-access system using this decorrelator is given. A delay-division multiplexing scheme based on this concept is also proposed.

1 Introduction

In a multiuser communications environment, the detector receives the sum of the signals transmitted by all the users. A code-division multiple-access (CDMA) system using conventional detection separates the signals and extracts the desired information symbols by matched filtering the received signal using a copy of the desired user's signature waveform. Multiuser detection typically relies on knowledge of *all* the user's parameters, such as signature waveform, delays, and sometimes received energy. Such information is often not known even in a friendly communications network.

Much research has focused on finding algorithms that perform simultaneous detection and parameter estimation in CDMA systems [1, 2]. These efforts have assumed that the signature sequences of all the users are known. Several works have considered detection of a single user, knowing that user's parameters but not those of other users [3, 4]. Some recent effort has focused on identifying a single new-user on a channel and determining its parameters, including the signature waveform, assuming knowledge of all other users' parameters [5, 6, 7]. One algorithm has been proposed that does not require knowledge of any parameters, including the signature sequences, these parameters being found iteratively [8]. This algorithm suffers from complexity exponential in the number of users and can also result in local-minimum solution. No low-complexity algorithm has been proposed that can detect all signals without any knowledge. This paper is the first step in that direction.

We propose a linear scheme to separate asynchronous users' signals without knowledge of *any* of the users' parameters, not even the signature sequence, by using fractionally spaced samples of the chips. No training sequence is necessary; the only necessary parameter is the oversampling rate. This is accomplished by inverting the "filter" imposed on the signals in

the delay operation. A zero-forcing equalizer (decorrelator) can be implemented such that all signals that have sufficiently different delays, relative to the oversampling rate, can be perfectly reconstructed in a noiseless environment. In the noisy case, a large received energy is required to provide sufficient signal-to-noise ratio at each sample.

This receiver can be implemented as a front-end to a large CDMA system to separate users with similar delays, which can then be demodulated using another (single-user or multiuser) detector. With sufficient oversampling, all users can be separated, and the information symbols can be found as in any single-user system. The proposed method can also be used in a small, well synchronized system to accomplish multiple-accessing in itself, i.e., without any spreading required. Each user can be assigned a different delay instead of a different code.

2 System Model

Consider a system composed of K users, labeled $k = 1, \ldots, K$, each transmitting binary data $b_k^{(i)} \in \{-1, 1\}$ for $i = 0, \ldots, M-1$, where M is the frame size. Each signal is modulated by a signature sequence $\{c_k(n) \in \{-1, 1\}\}_{n=0}^{N-1}$, where N is the length of the sequence, or processing gain, of the system. The baseband equivalent continuous time received signal from user k can be written

$$y_k(t) = A_k \sum_{i=0}^{M-1} b_k^{(i)} \sum_{n=0}^{N-1} c_k(n) g(t - nT - iNT - \tau_k), \tag{1}$$

where A_k is the (perhaps complex) amplitude of user k'th signal, T is the chip duration (the information symbol period divided by the processing gain) common to all users, τ_k is the relative delay of user k, and $g(t)$ is a unit square pulse of duration T. An arbitrary pulse shape which includes the effects of transmit, channel, and receive filters will be considered in Section 3.1. The signal received from user k can be sampled P times per chip giving the discrete time signal

$$\tilde{y}_k(m) = A_k \sum_{i=0}^{M-1} b_k^{(i)} \sum_{n=0}^{N-1} c_k(n) \tilde{g}(m - \alpha_k - P(n + iN + \beta_k)); \quad m = 0, \ldots, (M+1)NP - 1, \tag{2}$$

where $\tilde{g}(m) = g(mT/P)$ and $\lfloor \tau_k P/T \rfloor = \alpha_k + P\beta_k$, for integer valued α_k and β_k and $0 \leq \alpha_k < P$. Note that the vector $\tilde{g}$ is only nonzero for P samples, specifically,

$$\tilde{g}(m) = \begin{cases} 1 & 0 \leq m < P - 1, \\ 0 & \text{otherwise.} \end{cases} \tag{3}$$

Let the received signal be

$$\tilde{y}(m) = \sum_{k=1}^{K} \tilde{y}_k(m) + \nu(m); \quad m = 0, \ldots, (M+1)NP - 1, \tag{4}$$

where $\nu(m)$ is an additive Gaussian noise sequence with zero mean and covariance matrix $\sigma^2 I$, where I is the $(M+1)NP \times (M+1)NP$ identity matrix. The signal $\tilde{y}(m)$ is available to the decorrelator for processing.

3 Decorrelator

The goal of a decorrelator is to separate signals that have been overlapped. The decorrelator developed in this section performs this operation by inverse filtering the equivalent linear filter imposed on the various signals through delay. Consider the noiseless case first. Let us write the received signal $\tilde{y}(m)$ in an $(M+1)N \times P$ matrix, filling row by row. Then, column j is given by a vector $\underline{Z}^{(j)}$ with elements

$$Z^{(j)}(\ell) = \tilde{y}(\ell P + j); \quad \ell = 0, \ldots, (M+1)N - 1, \; j = 0, \ldots, P-1. \tag{5}$$

We can now write

$$Z^{(j)}(\ell) = \sum_{k=1}^{K} \sum_{i=0}^{M-1} \sum_{n=0}^{N-1} A_k b_k^{(i)} c_k(n) \tilde{g}(j - \alpha_k - P(n + iN + \beta_k - \ell)). \tag{6}$$

Partitioning the set of users according to their delay, we write $\{k^{(q)}\}$ as the set of all users with $\alpha_k = q$. Then,

$$\begin{aligned} Z^{(j)}(\ell) = & \sum_{\{k^{(0)}\}} \sum_{i=0}^{M-1} \sum_{n=0}^{N-1} A_k b_k^{(i)} c_k(n) \tilde{g}(j - P(n + iN + \beta_k - \ell)) + \\ & \sum_{\{k^{(1)}\}} \sum_{i=0}^{M-1} \sum_{n=0}^{N-1} A_k b_k^{(i)} c_k(n) \tilde{g}(j - 1 - P(n + iN + \beta_k - \ell)) + \ldots + \\ & \sum_{\{k^{(P-1)}\}} \sum_{i=0}^{M-1} \sum_{n=0}^{N-1} A_k b_k^{(i)} c_k(n) \tilde{g}(j - (P-1) - P(n + iN + \beta_k - \ell)). \end{aligned} \tag{7}$$

Clearly all $c_k(n)\tilde{g}(j - \alpha_k - P(n + iN + \beta_k - \ell))$ are equal for $j = \alpha_k, \ldots, \alpha_k + P - 1$. Let the P isolated terms in the sum in (7) be given the notation $Y^{(q)}(\gamma, \ell)$, $q = 0, \ldots, P-1$, where $\gamma = 0$ indicates $g(.)$ is nonzero for $(n + iN + \beta_k - \ell) = 0$ and $\gamma = -1$ indicates $g(.)$ is nonzero for $(n + iN + \beta_k - \ell) = -1$. The superscript q refers to the set of users $\{k^{(q)}\}$. We can now write

$$\begin{aligned} Z^{(0)}(\ell) &= Y^{(0)}(0,\ell) + Y^{(1)}(-1,\ell) + \ldots + Y^{(P-1)}(-1,\ell), \\ Z^{(1)}(\ell) &= Y^{(0)}(0,\ell) + Y^{(1)}(0,\ell) + Y^{(2)}(-1,\ell) + \ldots + Y^{(P-1)}(-1,\ell), \\ &\vdots \\ Z^{(P-2)}(\ell) &= Y^{(0)}(0,\ell) + Y^{(1)}(0,\ell) + \ldots + Y^{(P-2)}(0,\ell) + Y^{(P-1)}(-1,\ell), \\ Z^{(P-1)}(\ell) &= Y^{(0)}(0,\ell) + Y^{(1)}(0,\ell) + \ldots + Y^{(P-1)}(0,\ell). \end{aligned} \tag{8}$$

Now the simplification comes in noting that the $(M+1)N$-vector $\underline{Y}^{(q)}(-1)$ over ℓ is equal to $\Pi \underline{Y}^{(q)}(0)$, where Π is a matrix of all zeros except for the first lower diagonal. We now write the system of equations in matrix form as

$$\underline{Z} = \begin{pmatrix} \underline{Z}^{(0)} \\ \underline{Z}^{(1)} \\ \vdots \\ \underline{Z}^{(P-1)} \end{pmatrix} = \begin{pmatrix} I & \Pi & \Pi & \ldots & \Pi \\ I & I & \Pi & \ldots & \Pi \\ I & I & I & \ldots & \Pi \\ & & \vdots & & \\ I & I & I & \ldots & I \end{pmatrix} \begin{pmatrix} \underline{Y}^{(0)}(0) \\ \underline{Y}^{(1)}(0) \\ \vdots \\ \underline{Y}^{(P-1)}(0) \end{pmatrix},$$

$$= \mathbf{A}\underline{Y}. \tag{9}$$

The matrix I is the $(M+1)N \times (M+1)N$ identity matrix. By inspection, the determinant of $\mathbf{A}$ for any $(M+1)N$ and P is found to be unity, and it is therefore always invertible.

A simple exact solution for $\underline{Y}$ is found by simply inverting $\mathbf{A}$ and left-hand multiplying the vector $\underline{Z}$ by $\mathbf{A}^{-1}$. This operation yields a decorrelator for the sum signals $\underline{Y}$. In the noiseless case, we are able to recover exactly each users that has a delay α_k which is unique to that user. Let user k have such a unique delay α_k. Then, clearly, for any value of $j = \alpha_k, \ldots, \alpha_k + P - 1$,

$$Y^{(\alpha_k)}(0,\ell) = \sum_{i=0}^{M-1} \sum_{n=0}^{N-1} A_k b_k^{(i)} c_k(n) \tilde{g}(j - \alpha_k - P(n + iN + \beta_k - \ell)) = \tilde{y}_k(P\ell). \tag{10}$$

Thus the result of the decorrelation algorithm is to extract the user signal sampled only once each chip. Note that the structure of the signals in (2), i.e., the CDMA encoding, is not necessary for the success of this decorrelator. Any asynchronous system of digital signals with rectangular pulses can be decorrelated in this fashion. This point is elaborated further in Section 3.3

In the noisy case, the system of equations becomes

$$\underline{Z} = \mathbf{A}\,\underline{Y} + \underline{\nu}. \tag{11}$$

where $\underline{\nu}$ is a Gaussian vector with covariance matrix $\sigma^2 I$. The zero-forcing filter is the matrix $\mathbf{A}^{-1}$, yielding a soft decision

$$\hat{\underline{Y}} = \underline{Y} + \mathbf{A}^{-1}\underline{\nu} \tag{12}$$

which can be thresholded at zero for a hard decision. Unfortunately, the decorrelating operation enhances the noise by multiplying it by the matrix $\mathbf{A}^{-1}$. The noise covariance function becomes $\sigma^2 \mathbf{A}^{-1}(\mathbf{A}^{-1})^T$. A minimum mean squared error (MMSE) filter of the form $(\mathbf{A} + I\sigma^2)^{-1}$ could be used instead.

Once the signals have been separated, the information sequence for the desired user can be recovered, either before or after thresholding, by simply correlating with the expected signature sequence. If the signature sequences are not known, a single-user synchronization method must be used to find the delay β_k. Once the delay is known, the signature sequence can then easily be extracted and the data recovered.

3.1 Filtered Input

If the channel is not ideal, or if the signal has undergone some pulse shaping, our method may still be used if an inverse filter can be applied to the received signal before decorrelation. Suppose a filter with impulse response $h(t)$, due to the transmit filter, channel response, and receive filter, is applied to the continuous time received signal $y_k(t)$ in (1). The discrete-time oversampled signal $\tilde{y}_k(m)$ can then be given by

$$\tilde{y}_k(m) = A_k \sum_{i=0}^{M-1} b_k^{(i)} \sum_{n=0}^{N-1} c_k(n) \tilde{h}(m - \alpha_k - P(n + iN + \beta_k)); \quad m = 0, \ldots, (M+1)NP, \tag{13}$$

where $\tilde{h}(m) = \int_0^T h(t - mT/P)dt$, $\forall m$. The received signal is then given as in (4). If the filter $\tilde{h}$ can be inverted, then the inverse filter is applied to the received signal before decorrelation. The inverse filter $f(m)$ has the property that

$$g(m) = \sum_{j=0}^{(M+1)NP} \tilde{h}(j) f(m - j) \tag{14}$$

The signal input to the decorrelator after inverse filtering is exactly the same as if no filtering was performed. How this decorrelator may be applied to systems with noninvertable filters is the subject of current research.

3.2 Frame Synchronization Requirement

This decorrelator relies on the fact that all users transmit a finite frame of length M, and that all such frames are to a certain extent synchronized. The method as described above will not work for a continuous stream of data. If users with delay $\alpha_k > 0$ do not all have zeroes (no transmission) at the beginning of the frame, they will carry a fixed DC bias on all chip estimates. Since the information contains no DC component, this bias can be filtered in each frame if the frame is long enough. With this adjustment, the decorrelator can be used on a stream of data, processed one frame at a time. The required signal-to-noise ratio (SNR) will be shown to increase with the frame length M. Therefore, a trade-off exists in making the frame length M long enough to estimate the bias, yet as short as possible to minimize the noise.

3.3 Delay Division Multiple Access

The decorrelator described above allows asynchronous signals added together to be separated at the output. This idea can be used to design a type of multiplexing scheme, given the name delay division multiple-access (DDMA), that relies exclusively on the asynchronous nature of the user messages to differentiate between them. Each user can transmit a signal uncoded (in the CDMA sense) and if the oversampling rate is sufficient, all users can be isolated. This scheme allows an arbitrary number of users to occupy the same bandwidth as would be necessary for a single user, as long as the bandlimiting filter can be inverted at the receiver. The trade-off of using this scheme is therefore a severe increase in the required SNR in exchange for a significant decrease in the necessary bandwidth, compared to a time-division multiple-access (TDMA) system or any spread spectrum multiple-access system. The DDMA scheme is not altogether different from TDMA; each user is still assigned to transmit within a slot. The difference is that in DDMA the slots can overlap and, as long as the pulseshaping filter can be inverted, this requires less bandwidth. Note that to accomplish this type of decorrelation on a CDMA signal, the oversampling applies to the *chip* period while for the DDMA system the oversampling applies to the *symbol* period, and it therefore becomes more practical to oversample by orders of tens or hundreds. In the following section, a performance evaluation is given for the CDMA and DDMA systems considered.

4 Performance Analysis

Two kinds of problems can arise with this detector. If the delays are not distinct, signals overlap causing what we call a collision. In the noisy case, bit errors can also occur which will degrade the system performance even in the complete absence of collisions. The probability of a collision can be calculated as the probability that any other user, or any combination, collide with a given user. If the delays are uniformly distributed in $[0, TN]$ and independent of each other, the probability of two users colliding $\Pr(\alpha_j = \alpha_k)$ is given by $1/P$, which gives the

overall average probability of a collision as

$$P_c = 1 - \left(1 - \frac{1}{P}\right)^{K-1} \tag{15}$$

In the presence of noise, the performance of the decorrelator for unique-delay signals depends entirely on the values of the diagonal of the matrix $\mathbf{A}^{-1}(\mathbf{A}^{-1})^T$, i.e., the variance of the equivalent noise, which depend on the product $(M+1)N$ and on P. The probability of error is independent of the energies of the other users since true decorrelation is accomplished (this is obviously not true if the delays are not unique.) The diagonal of $\mathbf{A}^{-1}(\mathbf{A}^{-1})^T$ is given by

$$(\ 1\ \ 3\ \ \ldots\ \ (2(M+1)N-1)\ \mid\ 2\ \ 4\ \ \ldots\ \ 2(M+1)N\ \mid\ \ldots\ \mid\ 2\ \ 4\ \ \ldots\ \ 2(M+1)N\ )^T \tag{16}$$

where there are $P(M+1)N$ vector entries. The average probability of a chip error for user $k=1$ is then given by

$$P_{e,1} = \frac{1}{MN} \sum_{j=1}^{MN} Q\left(\frac{A_1}{\sigma}\sqrt{\frac{1}{2j-1}}\right) \tag{17}$$

while the error probability for a chip of any other user can be written

$$P_{e,k} = \frac{1}{MN} \sum_{j=1}^{MN} Q\left(\frac{A_k}{\sigma}\sqrt{\frac{1}{2j}}\right); \quad k = 2, \ldots, K. \tag{18}$$

Intuitively, it is understandable why the probability of error is slightly less for user $k=1$ since during part of that transmission there is no interference. Note that the probability of error for each chip is not the same. The first few chips on the frame experience much lower variance then the latter.

Shown in Figure 1 is the average probability of a chip error as a function of the input sample signal-to-noise ratio for unique-delay users. For $P=1$, the $\mathbf{A}$ matrix is just an identity matrix for any value of $(M+1)N$. It is evident that the penalty due to decorrelation for $P>1$ is more dependent on the product $(M+1)N$ than the oversampling rate P.

Let us now calculate the error probability for a CDMA system with no collisions and perfect knowledge of the signature sequences. Let the symbol energy be given by E_k. Each information symbol is composed of N chips, so the symbol energy can be written as $E_k/P = A_k N$, since each chip is sampled P times. The noise variance varies from chip to chip and symbol to symbol, giving an average error probability of

$$P_{e,k} = \frac{1}{M} \sum_{i=0}^{(M-1)} Q\left(\frac{E_k}{P\sigma}\sqrt{\sum_{j=1}^{N} \frac{1}{2(j+iN)}}\right); \quad k = 2, \ldots, K \tag{19}$$

for $k = 2, \ldots, K$, with the obvious change in the noise variance for $k=1$. A DDMA system is equivalent to a CDMA system with $N=1$.

Assuming collisions do occur, the DDMA system is severely limited by the collision probability of (15). The CDMA system on the other hand, reverts back to a multiuser system with the multiple-access interference being only from those users with the same delay. Knowledge

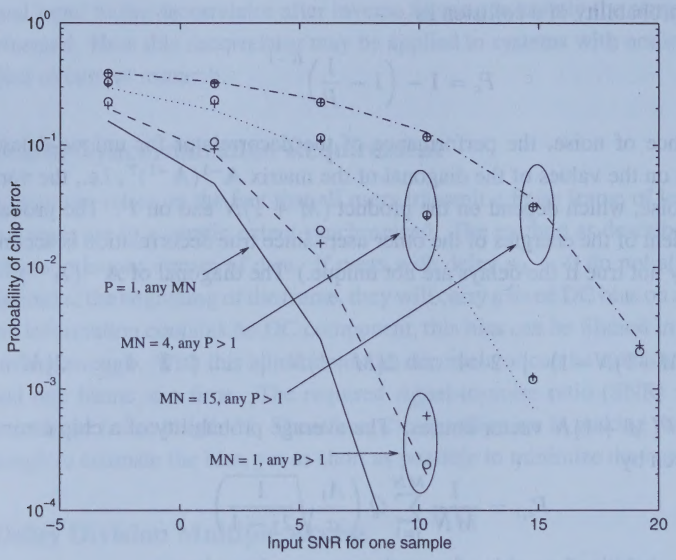


Figure 1: Probability of chip error as the signal-to-noise ratio increases. For the cases with $P > 1$, the lines, +s, and os represent $P = 2$, $P = 3$, and $P = 6$, respectively.

of the signature sequences is required in this case. If collisions are due to random transmit times, then the error probability can be calculated as

$$P_{e,k} = \sum_{d_1+2d_2+\ldots+Kd_K=K, d_1+\ldots,d_K \leq P} p_{d_1,d_2,\ldots,d_K} \sum_{k=1}^{K} k d_k(P_k)/K \tag{20}$$

where d_k is the number of distinct delays with exactly a k-users overlap, i.e., $p_{d_1,d_2,\ldots,d_K}$ is the joint probability that there are d_1 delays occupied by a single users, d_2 delays occupied by 2 users, *etc.*. P_k is the probability of error for a k-user synchronous CDMA system. The probability $p_{d_1,d_2,\ldots,d_K}$ can be found by an application of the birthday problem, as in collision-based communication systems [9]. This gives

$$p_{d_1,d_2,\ldots,d_K} = \frac{K!P!}{2!^{d_2}3!^{d_3}\cdots K!^{d_K} d_1!d_2!\cdots d_K!(P-K+(K-1)d_K+\cdots+2d_3+d_2)!P^K}. \tag{21}$$

The probability of error for a synchronous CDMA system composed of k users with equal energies E and crosscorrelations equal to ρ can be written as

$$P_k = \frac{1}{2^k M} \sum_{n=0}^{k-1} \binom{k-1}{n} \sum_{i=0}^{(M-1)} Q\left(\frac{\sqrt{E}(1+\rho(2n-k+1))}{P\sigma} \sqrt{\sum_{j=1}^{N} \frac{1}{2(j+iN)}} \right), \tag{22}$$

where $Q(x) = \int_x^\infty e^{-t^2/2}/\sqrt{2\pi}\, dt$. A conventional matched filter detector is assumed.

Substituting (21) and (22) into (20), the average error probability can be calculated and is plotted in Figure 2. There is obviously a strong SNR penalty in oversampling. Yet, when the number of users is large ($K = 5$ or 7), the system with smaller P hits a probability of error

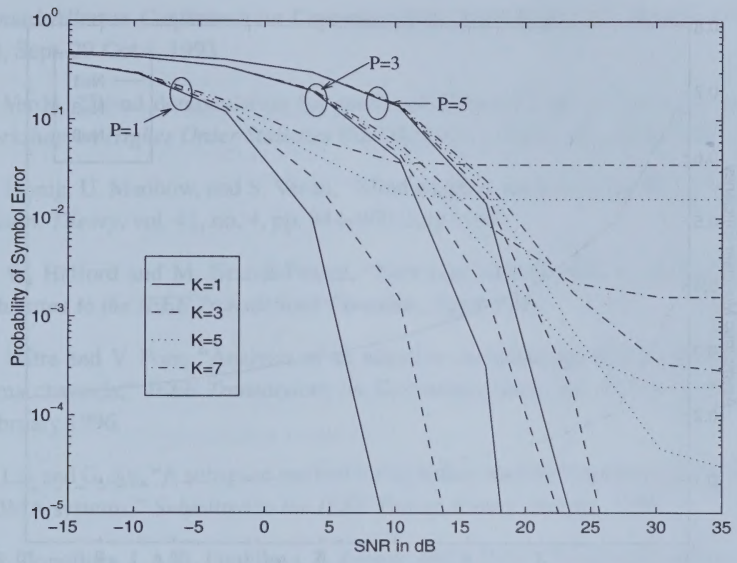


Figure 2: Probability of error for a CDMA system with collisions resolved using a matched-filter receiver.

floor which is higher than the system with a larger oversampling P. This figure uses a constant $\rho = 0.2$.

If there are more users on the system than samples per chip, there will always be collisions. In such a case, if the system is very well synchronized, a system can be designed such that there are less than N users overlapping in any "delay", for each delay. Since each output of the decorrelator yields overlapped signals that are synchronous to each other, the N or less users can be assigned orthogonal signatures. In this way, NP users can simultaneously transmit with absolutely no multiple-access interference, hence no near-far problem. The probability of error for each of the NK users in this case is given by (19).

The asymptotic multiuser efficiency (AME), as defined by Verdú in [10], can be given for the CDMA blind decorrelator with no collisions as

$$\frac{1}{P}\sqrt{\frac{1}{N}\sum_{j=1}^{N}\frac{1}{2(j+MN)}} \tag{23}$$

The AME is calculated via an infimum, which is equivalent to considering the worst case situation, which occurs for the last symbol in the frame. The AME is plotted for different values of M and N, for $P = 1$, in Figure 3. If collisions are present, the AME depends on the AME of the single-user or multiuser detector used on each delay. The worst case obviously occurs when all users have the same delay, so the AME for a system with this decorrelator front-end is given by the AME of the CDMA detector with the noise variance adjusted for the effect of this front-end filter, and a factor of $1/P$ due to oversampling.

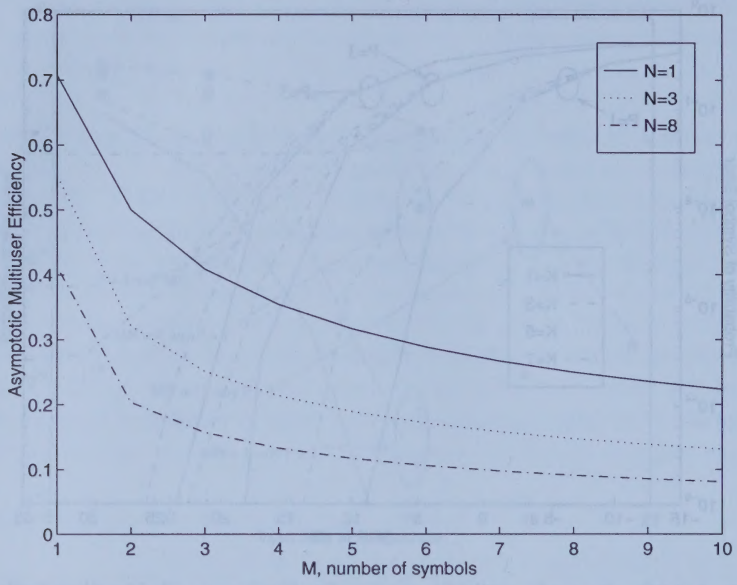


Figure 3: AME for a CDMA system with $P = 1$ and no collisions.

5 Conclusions

A blind decorrelator is presented which uses the different delays of the users to differentiate amongst them. It is shown that a frame of bits can be demodulated into groups of users all sharing the same relative delay. If a filter is imposed on the signal, as long as it is invertible the method is still valid. In a CDMA environment, each group can then be demodulated independently. A system named delay-division multiple-access can be devised in which exactly one user is assigned to each delay. This scheme has bandwidth advantages over spread spectrum schemes.

Unfortunately, the signal-to-noise ratio penalty for using this blind decorrelator is severe and increases with the size of the frame and the oversampling rate. A probability of error advantage of using this method for CDMA systems using conventional detection does exist for a large number of users. The asymptotic multiuser efficiency is shown to degrade rapidly as the size of the frame increases.

References

[1] R. S. Z. Xie, C.K. Rushforth and T. Moon, "Joint signal detection and parameter estimation in multiuser communications," *IEEE Transactions on Communications*, vol. 41, no. 7, pp. 1208–1215, August 1993.

[2] A. Radović and B. Aazhang, "Iterative algorithms for joint data detection and delay estimation for code division multiple access communications systems," in *Proc. of the 31st*

Annual Allerton Conference on Communication, Control and Computing, (Monticello, IL), Sept. 29-Oct.1, 1993.

[3] S. Verdú, "Blind demodulation for multiuser channels," in *Proceedings of the 1995 Workshop on Higher Order Statistics IEEE/ATHOS*, (Aiguavabla, Spain), June 1995.

[4] M. Honig, U. Madhow, and S. Verdú, "Blind adaptive multiuser detection," *IEEE Trans. Inform. Theory*, vol. 41, no. 4, pp. 944–960, July 1995.

[5] K. W. Halford and M. Brandt-Pearce, "New-user identification in a CDMA system," *Submitted to the IEEE Transactions Commun.*, April 1996.

[6] U. Mitra and V. Poor, "Analysis of an adaptive decorrelating detector for synchronous cdma channels," *IEEE Transactions on Communications*, vol. 44, no. 2, pp. 257–268, February 1996.

[7] H. Liu and G. Xu, "A subspace method for signature waveform estimation if synchronous CDMA systems," *Submitted to the IEEE Transactions Commun.*, 1995.

[8] J. R. Fonollosa, J. A. R. Fonollosa, Z. Zvonar, and J. Vidal, "Blind multiuser identification and detection in CDMA systems," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing (ICASSP)*, (Detroit, MI), May 1995.

[9] K. W. Halford and M. Brandt-Pearce, "Performance of a multistage multiuser detector for a frequency hopping multiple-access system," in *Proceedings of the 30th Annual Conference on Information Sciences and Systems*, (Princeton, NJ), March 1996.

[10] S. Verdú, "Optimum multiuser asymptotic efficiency," *IEEE Trans. Commun.*, vol. COM-34, no. 9, pp. 890–897, Sept, 1986.

Multiuser Decision-Feedback Receivers for the General Gaussian Multiple-Access Channel

Tommy Guess and Mahesh K. Varanasi †
Department of Electrical and Computer Engineering
University of Colorado, Boulder, CO 80309

Abstract

This paper considers multiple users accessing a common additive white Gaussian noise channel. The users are assigned signature waveforms, their signaling is synchronous, and hence the equivalent discrete-time model is a general Gaussian multiple-access channel (GMAC). A new class of receivers is proposed. The receivers within this class feature low-complexity detectors and allow the users to choose single-user codes independently of each other. The latter is accomplished by decoding the users sequentially, removing the contribution of the previously decoded user at each stage using feedback. These decision-feedback receivers (DFR) provide a continuum between single-output onion peeling and the Cholesky DFR. The latter uses the detector to remove inter-user interference, while the former does not. For a given set of users with fixed powers and signature waveforms, the receiver within this class that simultaneously maximizes the capacity of each user is found. This is the capacity-maximizing DFR.

1 Introduction

The capacity region for the conventional multiuser Gaussian channel is well known [1]. Although the boundary points of this capacity region can be achieved by random codes, the decoding complexity for such codes is exorbitant. Meanwhile, research into the design of joint codebooks for the users that allow them to approach the boundary of the capacity region is still in its fledgling stages. Another approach to this problem is to decode the users sequentially by means of *onion peeling* [1] [2] where the first user regards the other users as noise. The decoded symbols of the first user can be removed from the received codeword. If the symbols are decoded correctly, the result is that the first user is effectively removed from the system. This procedure is repeated for the second user and sequentially until the last user is all that is left to be decoded. This approach allays the primary difficulty of multiuser coding since it breaks the multiuser problem into several single-user problems, and single-user coding theory is well established. Such a strategy can achieve the vertices of the capacity region.

In this paper we apply the concept of sequentially decoding the users to the situation where they are allowed to use spreading waveforms. Spreading refers to the bandwidth

† This work was supported in part by NSF grant NCR-9406069.

expansion needed to accommodate the signature waveforms of the users in a code-division multiple-access system so as to provide some separation of the users. Whereas, if all of the users have the same signature waveform, then there is no spreading and error-control coding alone must be used to separate the users. When the users operate synchronously, the equivalent discrete-time model is the general Gaussian multiple-access channel (GMAC), a channel whose capacity region is given in [3]. This is a linear vector channel that is defined by a correlation matrix that contains the correlations of the users' signature waveforms with one another. When the users all have the same signature waveform (no spreading), this general GMAC reduces to the conventional GMAC [1] [3], and when the users have orthogonal signature waveforms (maximum spreading) this channel is simply a bank of single-user channels. In [4] it was shown that when the users are energy-weighted root-mean-squared (EW-RMS) bandlimited, the total capacity of the channel is in general maximized with signature waveforms that are neither orthogonal nor identical. This is an incentive for considering the general GMAC.

We want our receiver to have several properties. First, the detector should have a low complexity, preferably linear in the number of users, and it must be applicable to cases where the correlation matrix of the users is singular. Second, like onion peeling, it should decode the users sequentially (each user is decoded based on a single output of the vector channel) so that the users can choose single-user codes independently of each other. We propose a class of decision-feedback receivers (DFR) that satisfy the desired requirements. When the users' powers and signature waveforms are fixed, the receivers within this class cover the range between, and including, single-output onion peeling (SOOP) and the Cholesky-DFR (C-DFR) [5]. By SOOP we mean onion peeling that is applied to the general GMAC with each user being decoded based on a single output of the vector channel. For example, the first user is decoded based on the sampled outputs of the first matched filter as opposed to being decoded based on the sampled outputs of *all* the matched filters. The detector portion of the C-DFR uses the signature waveforms to separate the users by removing inter-user interference, but for correlated users the effective powers of the users can be severely reduced. SOOP, on the other hand, does not use the signature waveforms to reduce inter-user interference, and at high signal-to-noise ratios (SNR) this can significantly increase the effective noise powers seen by the users. General DFRs provide a continuum between these two extremes by varying the extent to which the receiver uses the detector to remove inter-user interference. The receiver that simultaneously maximizes the capacities of the users is found; it is the capacity-maximizing DFR (CM-DFR). In channels that emphasize spreading, the CM-DFR it looks like the C-DFR, but in regions where spreading is not important it behaves like the SOOP receiver.

The remainder of this paper is organized as follows: Section 2 introduces the general GMAC and shows its relation to synchronous CDMA when the users are bandlimited. Section 3 considers DFRs and develops the CM-DFR.

2 The System

Consider a PAM, synchronous, Gaussian CDMA channel where users are assigned signature waveforms that are time-limited to $[0, T]$. The signal that is received at some central location is modeled as the sum of M transmitting users and white Gaussian noise; it is given by

$$r(t) = \sum_{k=1}^{N} \sum_{i=1}^{M} b_i(k) u_i(t - kT) + n(t), \tag{1}$$

where $\{b_i(k)\}_k$ is the sequence of information symbols sent by the i^{th} user, $u_i(t)$ is the signature waveform of the i^{th} user, and $n(t)$ is an additive white Gaussian noise process with a power-spectral density of $N_0/2$. If each user's input waveform is power constrained to be w_i, then without loss of generality it can be assumed that $\frac{1}{NT} \sum_{k=1}^{N} b_i^2(k) = w_i$ for each i, and that the energy of the i^{th} user's waveform is $1 = \int_0^T u_i^2(t) dt$. Define the elements of the correlation matrix, $\mathbf{R}$, associated with a given set of users as

$$\mathbf{R}_{mn} = \int_0^T u_m(t) u_n(t) dt, \tag{2}$$

so that the each diagonal element of $\mathbf{R}$ is unity, and let $\mathbf{W}$ be a diagonal matrix with $\{w_1, w_2, \ldots, w_M\}$ as its diagonal. The received signal is input into a bank of filters that are matched to the M signature waveforms of the users, and the outputs of these filters are sampled at integer multiples of T to yield a set of sufficient statistics for the users' symbols. The channel is memoryless so it can be considered on a per-symbol basis. Letting $\mathbf{r}(k)$ denote an M-dimensional vector containing these sufficient statistics at time k, we have the following general GMAC

$$\mathbf{r}(k) = \mathbf{R}\mathbf{b}(k) + \mathbf{n}(k), \tag{3}$$

where $\mathbf{b}(k)$ is the vector of information symbols of the users and $\{\mathbf{n}(k)\}$ is a sequence of independent, identically-distributed(i.i.d.), zero-mean, Gaussian random vectors, each with covariance $\frac{N_0}{2}\mathbf{R}$.

3 Decision-Feedback Receivers for the GMAC

Consider a receiver structure where $\mathbf{r}(k)$ is input into a linear feedforward filter, the matrix $\mathbf{F}$. The first user is decoded based on the output $\text{row}_1(\mathbf{F})\mathbf{r}(k)$ (where $\text{row}_1(\mathbf{F})$ denotes the first row vector of $\mathbf{F}$), and these symbol decisions are subtracted from the other users after being fed back through a linear feedback filter, the matrix $\mathbf{B}$. Similarly, the second user is then decoded and the symbol decisions of this user are subtracted from users 3 to M after being fed back through $\mathbf{B}$. This process is repeated for users 3 up to M. Since we are considering capacity as the performance measure, we assume that all users are decoded correctly. Then, if $\mathbf{y}_i(k)$ denotes the effective channel of the i^{th} user, we have that

$$\mathbf{y}(k) = (\mathbf{F}\mathbf{R} - \mathbf{B})\mathbf{b}(k) + \tilde{\mathbf{n}}(k), \tag{4}$$

where $\{\tilde{\mathbf{n}}(k)\}$ is a sequence of i.i.d., zero-mean, Gaussian random vectors with covariance $\frac{N_0}{2}\mathbf{F}\mathbf{R}\mathbf{F}^{\mathrm{T}}$, where the superscript T means transpose. Of course $\mathbf{B}$ is strictly lower

triangular since only the decoded symbols of previous users are fed back. So for the i^{th} user we have

$$\mathbf{y}_i(k) = \mathbf{K}_{ii}\mathbf{b}_i(k) + \sum_{j \neq i} \mathbf{K}_{ij}\mathbf{b}_j(k) - \sum_{j=1}^{i-1} \mathbf{B}_{ij}\mathbf{b}_j(k) + \tilde{\mathbf{n}}_i(k), \tag{5}$$

where $\mathbf{K} = \mathbf{FR}$. When the input symbol distributions of the users are Gaussian, then it is readily seen that from the perspective of the i^{th} user this is a single-user, discrete-time Gaussian channel where the effective noise power is the sum of the system noise, $\tilde{\mathbf{n}}_i(k)$, and the "noise" of the interfering users, $\sum_{j \neq i} \mathbf{K}_{ij}\mathbf{b}_j(k) - \sum_{j=1}^{i-1} \mathbf{B}_{ij}\mathbf{b}_j(k)$. Thus the capacity of the effective single-user channel in nats per second is

$$C_i = \frac{1}{2T} \log \left(1 + \frac{\mathbf{K}_{ii}^2 w_i}{\sum_{j \neq i} (\mathbf{K}_{ij} - \mathbf{B}_{ij})^2 w_j + \frac{N_0}{2T} (\mathbf{FRF}^{\mathrm{T}})_{ii}} \right). \tag{6}$$

The ratio within the large parentheses of (6) shall be referred to as the signal-to- interference ratio of user i (SIR_i) because it includes interference from both the noise source and from the other users.

3.1 The Orthogonal Decision-Feedback Receiver

The orthogonal DFR (O-DFR) is applicable whenever the users' signature waveforms are orthogonal. The correlation matrix, $\mathbf{R}$, is the identity matrix and the system reduces to M single-user channels. Letting the feedfoward filter be the identity matrix and the feedback filter be the zero matrix(i. e., there is no feedback) allows all users to achieve their single-user capacities, that is, the capacities they would achieve if no interfering users were present. Of course this ideal performance comes at the cost of bandwidth. Time-division multiple access (TDMA) is an example of an orthogonal signaling scheme. It is not, however, the most efficient from a bandwidth point of view. That is, there are orthogonal signature waveforms that require less EW-RMS bandwidth than TDMA [4]. The set of orthogonal signature waveforms that minimize the bandwidth over all possible sets of orthogonal signature waveforms will be referred to as optimum-orthogonal multiple access (OOMA). OOMA is not a subset of TDMA since the non-zero portions of the OOMA signature waveforms, in general, overlap each other. This point was overlooked in [5], where TDMA should be replaced by OOMA.

3.2 Single-Output Onion Peeling

Single-output onion peeling (SOOP) is simply the application of onion peeling to the general GMAC with all of the users considering only the sampled outputs of their respective matched filters. It is a scheme that makes no effort to remove the interference of the other users with the feedforward filter. So $\mathbf{F}$ is once again chosen to be the identity matrix, and the feedback filter, $\mathbf{B}$, is the strictly lower-triangular part of the correlation matrix, $\mathbf{R}$. The capacity of the i^{th} user is

$$C_i = \frac{1}{2T} \log \left(1 + \frac{w_i}{\sum_{j=i+1}^{M} \mathbf{R}_{ij}^2 w_j + \frac{N_0}{2T}} \right). \tag{7}$$

If $\mathbf{R} = \mathbf{I}$ then this is the same as the orthogonal case, but here $\mathbf{R}$ can even be such that every element is unity. The latter situation is ideal from the point of view of bandwidth conservation, but at high SNRs the interfering users significantly limit the capacity since their interference term dominates the denominator of the SIR_i.

3.3 The Cholesky Decision-Feedback Receiver

The Cholesky DFR (C-DFR) was introduced in [5] (there it was called the Cholesky sequential decoder, but the term sequential decoder has been abandoned to avoid confusion with sequential decoding which is a method of decoding on single-user channels). It is a valid decoder whenever the correlation matrix is positive definite, so it inherently requires all the signature waveforms be linearly independent. When this is true, the correlation matrix, $\mathbf{R}$, is positive definite, and a Cholesky decomposition can be used to write $\mathbf{R} = \mathbf{L}^{\mathrm{T}}\mathbf{L}$ where $\mathbf{L}$ is a lower triangular matrix. The feedforward filter is given by $\mathbf{F} = (\mathbf{L}^{\mathrm{T}})^{-1}$ and the feedback filter by $\mathbf{B} = \mathbf{L} - \mathrm{diag}(\mathbf{L})$, where the $\mathrm{diag}(\mathbf{A})$ operation yields a diagonal matrix whose diagonal elements are the same as those of $\mathbf{A}$. Note that the noise term, $\tilde{\mathbf{n}}$, is white with each element having a power of $N_0/2$, and that $\mathbf{K} = \mathbf{L}$; both are true because of the decorrelating effect of $\mathbf{F}$. The resulting expression for (5) is $\mathbf{y}_i(k) = \mathbf{L}_{ii}\mathbf{b}_i(k) + \tilde{\mathbf{n}}_i(k)$, and this is independent of interference from other users. The capacity of the i^{th} user is given as

$$C_i = \frac{1}{2T}\log\left(1 + \frac{\mathbf{L}_{ii}^2 w_i}{N_0/(2T)}\right). \tag{8}$$

The cost of removing inter-user interference is that the $\mathbf{L}_{ii}^2$ term is less than unity unless the i^{th} user is orthogonal to users $i+1$ to M; this is a property of the Cholesky decomposition. Thus there is a potential tradeoff between the ability of the feedfoward filter to remove the effects of the interfering users, and the simultaneous reduction in the effective energy of the user of interest.

Finally, it should be pointed out that the C-DFR is unique among DFRs (excluding the relatively uninteresting case of orthogonal signaling) in the sense that it is applicable even when the users' symbol distributions are not Gaussian. This is true because the decorrelating action of the feedforward filter means that once the interference of the previous users has been removed by feedback, there is no residual interference (from other users).

3.4 The Capacity-Maximizing Decision-Feedback Receiver

Motivated by the observations of the previous subsections, we wish to optimize the capacity of the i^{th} user over all possible choices for the feedforward and feedback filters when the order in which the users are decoded is given. This is done under the assumption that the users' powers are fixed, as are N_0 and T. Thus we seek to solve

$$\mathbf{F}_{\mathrm{opt}}, \mathbf{B}_{\mathrm{opt}} \in \arg\max_{\mathbf{F},\mathbf{B}}\{C_i\}, \tag{9}$$

where $\mathbf{F}$ and $\mathbf{B}$ are real $M \times M$ matrices with $\mathbf{B}$ being strictly lower triangular, and C_i given by (6). Because the logarithm is a monotonic increasing function and T is fixed,

the problem can be expressed equivalently as

$$\mathbf{F}_{\text{opt}}, \mathbf{B}_{\text{opt}} \in \arg\max_{\mathbf{F},\mathbf{B}} \left\{ \frac{\mathbf{K}_{ii}^2 w_i}{\sum_{j \neq i} (\mathbf{K}_{ij} - \mathbf{B}_{ij})^2 w_j + \frac{N_0}{2T} (\mathbf{F}\mathbf{R}\mathbf{F}^{\mathrm{T}})_{ii}} \right\}. \tag{10}$$

This objective function depends only the i^{th} rows of $\mathbf{F}$ and $\mathbf{B}$. This means that each row of $\mathbf{F}$ and $\mathbf{B}$ can be chosen to maximize the capacity of the corresponding user, and hence the capacities of all the users can be simultaneously maximized. Let $\mathbf{f}^{\mathrm{T}}$ denote the i^{th} row of $\mathbf{F}$ and β^{T} the i^{th} row of $\mathbf{B}$ so that (10) becomes

$$\mathbf{f}_{\text{opt}}, \beta_{\text{opt}} \in \arg\max_{\mathbf{f},\beta} \left\{ \frac{(\mathbf{f}^{\mathrm{T}} \text{col}_i(\mathbf{R}))^2 w_i}{(\mathbf{f}^{\mathrm{T}}\mathbf{R} - \beta^{\mathrm{T}}) \hat{\mathbf{W}}_i (\mathbf{R}\mathbf{f} - \beta) + \frac{N_0}{2T} \mathbf{f}^{\mathrm{T}}\mathbf{R}\mathbf{f}} \right\}, \tag{11}$$

where $\hat{\mathbf{W}}_i$ denotes $\mathbf{W}$ with the i^{th} diagonal element being set to zero, and $\text{col}_i(\mathbf{A})$ denotes the i^{th} column vector of $\mathbf{A}$. Since the numerator is independent of β, the best β can do to maximize the objective function is make the denominator as small as possible. It is readily seen that the nonzero elements of β_{opt} are the same as the first $i-1$ elements of $\mathbf{R}\mathbf{f}$ for any $\mathbf{f}$ that is used. That is, β_{opt} simply removes the interference from the previous users. Since, for any user i, $\mathbf{B}$ has removed the interference of users 1 to $i-1$, user i "looks" like a first user. Therefore, to facilitate notation, we will proceed by considering only the first user.

Let $\hat{\mathbf{W}}_{j:k}$ be $\mathbf{W}$ with the diagonal elements j to k set to zero. Thus we are left to find

$$\mathbf{f}_{\text{opt}} \in \arg\max_{\mathbf{f}} \left\{ \frac{\mathbf{f}^{\mathrm{T}}\mathbf{R}\hat{\mathbf{W}}_{2:M}\mathbf{R}\mathbf{f}}{\mathbf{f}^{\mathrm{T}}(\mathbf{R}\hat{\mathbf{W}}_1\mathbf{R} + \frac{N_0}{2T}\mathbf{R})\mathbf{f}} \right\}. \tag{12}$$

For the case where $\mathbf{R}$ is positive definite the term in the parentheses is positive definite and this maximization is solved via a change of variables to transform the problem into a Rayleigh-Ritz parameterization of the largest eigenvalue. First make a change of variables so that $\mathbf{k} = \mathbf{R}\mathbf{f}$, an invertible transformation by virtue of the invertibility of $\mathbf{R}$. The SIR of user 1 is now

$$\text{SIR}_1 = \frac{\mathbf{k}^{\mathrm{T}}\hat{\mathbf{W}}_{2:M}\mathbf{k}}{\mathbf{k}^{\mathrm{T}}(\hat{\mathbf{W}}_1 + \frac{N_0}{2T}\mathbf{R}^{-1})\mathbf{k}}. \tag{13}$$

Now let $\mathbf{J}^{\mathrm{T}}\mathbf{J}$ be the Cholesky decomposition of

$$\mathbf{S} \triangleq \hat{\mathbf{W}}_1 + \frac{N_0}{2T}\mathbf{R}^{-1} = \mathbf{J}^{\mathrm{T}}\mathbf{J}, \tag{14}$$

where $\mathbf{J}$ is lower triangular; $\mathbf{J}$ is invertible. Another change of variables with $\mathbf{s} = \mathbf{J}\mathbf{k}$ gives that

$$\text{SIR}_1 = \frac{\mathbf{s}^{\mathrm{T}}(\mathbf{J}^{\mathrm{T}})^{-1}\hat{\mathbf{W}}_{2:M}\mathbf{J}^{-1}\mathbf{s}}{\mathbf{s}^{\mathrm{T}}\mathbf{s}}. \tag{15}$$

This is maximized when $\mathbf{s}$ is any eigenvector corresponding to the largest eigenvalue of $(\mathbf{J}^{\mathrm{T}})^{-1}\hat{\mathbf{W}}_{2:M}\mathbf{J}^{-1}$, a diagonal matrix with only the 1,1 element being non-zero. The maximizing $\mathbf{s}$ is any non-zero multiple of the first column of the identity matrix. Hence we have that

$$\mathbf{f}_{\text{opt}} = \alpha\mathbf{R}^{-1}\text{col}_1\left(\mathbf{J}^{-1}\right), \tag{16}$$

where α is any non-zero constant, and that the resulting maximum SIR for user 1 is

$$\mathrm{SIR}_1 = \frac{w_1}{\mathbf{J}_{11}^2} = w_1 \left(\mathbf{S}^{-1}\right)_{11}. \tag{17}$$

The second equality follows from the relationship of the Cholesky decomposition to the Schur complement.

These results are extended in Appendix A to cover the case where $\mathbf{R}$ is only positive semi-definite. In Appendix B it is shown that, remarkably, this capacity-maximizing DFR (CM-DFR) is also a minimum mean-squared error (MMSE) DFR. That is, any $\mathbf{f}$ satisfying (16) also minimizes the MSE given by $\mathrm{E}\left\{(\mathbf{f}^{\mathrm{T}}\mathbf{R}\mathbf{b} + \tilde{\mathrm{n}}_1 - \mathrm{b}_1)^2\right\}$, and vice versa. A nice result of this equivalence is that the MMSE yields a very simple representation of the optimum feedforward and feedback filters (e.g., [7]),

$$\begin{aligned} \mathbf{F} &= \left(\mathbf{\Phi}^{\mathrm{T}}\right)^{-1} \mathbf{W}^{1/2} \\ \mathbf{B} &= \left(\mathbf{\Phi} - \mathrm{diag}\left(\mathbf{\Phi}\right)\right) \mathbf{W}^{-1/2}, \end{aligned} \tag{18}$$

where $\mathbf{\Phi}^{\mathrm{T}}\mathbf{\Phi} = \mathbf{W}^{1/2}\mathbf{R}\mathbf{W}^{1/2} + \frac{N_0}{2T}\mathbf{I}$, and $\mathbf{\Phi}$ is lower triangular.

3.5 Total Capacity of DFRs Compared

We now illustrate how the DFRs perform relative to each other. When the signature sequences are non-orthogonal and linearly independent, we can compare the total capacities of SOOP, the C-DFR, and the CM-DFR as the noise power varies. Examples can be seen in Figures 1 and 2, where the time interval, T, is chosen for each case so the the EW-RMS bandwidth [4] is 100 Hz. The correlation matrices for the two cases are

$$\mathbf{R} = \begin{bmatrix} 1.0 & 0.8 & 0.6 \\ 0.8 & 1.0 & 0.4 \\ 0.6 & 0.4 & 1.0 \end{bmatrix}, \quad \mathbf{R} \begin{bmatrix} 1.0 & 0.9 & 0.9 \\ 0.9 & 1.0 & 0.9 \\ 0.9 & 0.9 & 1.0 \end{bmatrix}. \tag{19}$$

In the first case all of the users have unit power, while in the second they have powers of 4, 2 and 1. The SNR is defined to be the SNR of the weakest user, i. e., $10\log\left(\frac{w_M}{N_0}\right)$. Note that at very high SNRs the performances of the CM-DFR and the C-DFR coincide. This happens because the noise power is very small compared to the power of interfering users, so the CM-DFR removes all interfering users by using a decorrelating feedforward filter. Similarly, at low SNRs the CM-DFR and SOOP have similar performances because allowing significant interference from other users degrades the SIR very little (since the SIR is dominated by the noise term) but also enables the effective power of the user of interest to be maximized. In these extreme regions it should be noticed that, in general, either the C-DFR or SOOP will not perform as well as the other. There are also regions in between the extremes where the CM-DFR performs significantly better than either the C-DFR or SOOP.

Summary

We have proposed a decision-feedback receiver structure that incorporates a low- complexity detector in conjunction with decision-feedback decoding for multiuser Gaussian

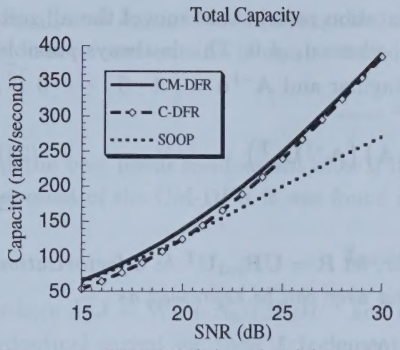


Figure 1: EW-RMS bandwidth is 100 Hz.

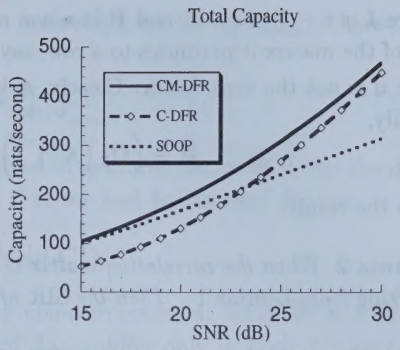


Figure 2: EW-RMS bandwidth is 100 Hz.

channels. Such receivers enable the users to incorporate error-control coding independently of each other using single-user codes. Single-output onion peeling (here applied to the general GMAC) and the C-DFR, two receivers in the literature that also use decision-feedback decoding, are special cases within this class of receivers. For a given correlation matrix we have derived the receiver from this class that simultaneously maximizes the capacity of each user, i. e., the CM-DFR.

Appendix

A Optimum DFR When R is Singular

Lemma 1 *Let* $\mathbf{R}$ *be a real,* $M \times M$*, normalized correlation matrix so that* $\mathbf{R}$ *is positive semi-definite and* $\mathrm{diag}(\mathbf{R}) = \mathbf{I}$*.* $\mathbf{R}$ *can be factored as*

$$\mathbf{R} = \mathbf{U}\mathbf{R}_{\mathrm{red}}\mathbf{U}^{\mathrm{T}} \triangleq \begin{bmatrix} 1 & 0 & \cdots & 0 \\ & & \mathbf{V} & \end{bmatrix} \mathbf{R}_{\mathrm{red}} \begin{bmatrix} 1 & \\ 0 & \\ \vdots & \mathbf{V}^{\mathrm{T}} \\ 0 & \end{bmatrix}, \tag{20}$$

where $\mathbf{V}$ *is an* $M-1 \times n$ *matrix and* $\mathbf{R}_{\mathrm{red}}$ *(the subscript* red *denotes the word reduced) is a* $n \times n$ *matrix with* n *representing the rank of* $\mathbf{R}$*. In addition, the ranks of* $\mathbf{U}$ *and* $\mathbf{R}_{\mathrm{red}}$ *are both* n*.*

Proof: A singular-value decomposition of the real symmetric $\mathbf{R}$ allows us to write

$$\mathbf{R} = \mathbf{U}_1 \mathbf{\Lambda} \mathbf{U}_1{}^{\mathrm{T}}, \tag{21}$$

with the dimensions of $\mathbf{U}_1$ and $\mathbf{\Lambda}$ being $M \times n$ and $n \times n$ respectively; both have a rank of n. Let $\mathbf{u}^{\mathrm{T}}$ be the first row of $\mathbf{U}_1$. We know that it cannot be the zero vector since the 1,1 element of $\mathbf{R}$ is unity. We let

$$\mathbf{A} = \begin{bmatrix} \mathbf{u} & \mathbf{P} \begin{bmatrix} \mathbf{0}^{\mathrm{T}} \\ \mathbf{I} \end{bmatrix} \end{bmatrix}, \tag{22}$$

where $\mathbf{I}$ is $n-1 \times n-1$, and $\mathbf{P}$ is a $n \times n$ permutation matrix that moves the all-zero row of the matrix it permutes to a row, say row k, where $\mathbf{u}_k \neq 0$. This is always possible since $\mathbf{u}$ is not the zero vector. Clearly, $\mathbf{A}$ is nonsingular and $\mathbf{A}^{-1}\mathbf{u} = \begin{bmatrix} 1 & 0 & \cdots & 0 \end{bmatrix}^{\mathrm{T}}$. Finally,

$$\mathbf{R} = \left(\mathbf{U}_1 \left(\mathbf{A}^{\mathrm{T}}\right)^{-1}\right)\left(\mathbf{A}^{\mathrm{T}}\mathbf{\Lambda}\mathbf{A}\right)\left(\mathbf{A}^{-1}\mathbf{U}_1{}^{\mathrm{T}}\right) \tag{23}$$

gives the result. □

Lemma 2 *When the correlation matrix is singular, let* $\mathbf{R} = \mathbf{U}\mathbf{R}_{\mathrm{red}}\mathbf{U}^{\mathrm{T}}$ *be a factorization resulting from* Lemma 1 . *Then the* SIR *of the first user can be expressed as*

$$\mathrm{SIR}_1 = w_1 \left(\mathbf{U}^{\mathrm{T}}\hat{\mathbf{W}}_1\mathbf{U} + \frac{N_0}{2T}\mathbf{R}_{\mathrm{red}}^{-1}\right)^{-1}_{11}. \tag{24}$$

Proof: Substitute the factorization of $\mathbf{R}$ into (12) to get that

$$\mathbf{f}_{\mathrm{opt}} \in \arg\max_{\mathbf{f}} \left\{ \frac{\mathbf{f}^{\mathrm{T}}\mathbf{U}(\mathbf{R}_{\mathrm{red}}\mathbf{U}^{\mathrm{T}}\hat{\mathbf{W}}_{2:M}\mathbf{U}\mathbf{R}_{\mathrm{red}})\mathbf{U}^{\mathrm{T}}\mathbf{f}}{\mathbf{f}^{\mathrm{T}}\mathbf{U}(\mathbf{R}_{\mathrm{red}}\mathbf{U}^{\mathrm{T}}\hat{\mathbf{W}}_1\mathbf{U}\mathbf{R}_{\mathrm{red}} + N_0/(2T)\mathbf{R}_{\mathrm{red}})\mathbf{U}^{\mathrm{T}}\mathbf{f}} \right\}. \tag{25}$$

The change of variables of $\mathbf{y} = \mathbf{R}_{\mathrm{red}}\mathbf{U}^{\mathrm{T}}\mathbf{f}$ leads to

$$\mathbf{y}_{\mathrm{opt}} \in \arg\max_{\mathbf{y}} \left\{ \frac{\mathbf{y}^{\mathrm{T}}(\mathbf{U}^{\mathrm{T}}\hat{\mathbf{W}}_{2:M}\mathbf{U})\mathbf{y}}{\mathbf{y}^{\mathrm{T}}(\mathbf{U}^{\mathrm{T}}\hat{\mathbf{W}}_1\mathbf{U} + N_0/(2T)\mathbf{R}_{\mathrm{red}}^{-1})\mathbf{y}} \right\}. \tag{26}$$

Even though the transformation by $\mathbf{U}$ is not invertible, it does not matter because this simply means that the set of $\mathbf{f}_{\mathrm{opt}}$ solutions is given by more than one basis vector.

Now factor $\mathbf{S}_{\mathrm{red}} \triangleq \mathbf{U}^{\mathrm{T}}\hat{\mathbf{W}}_1\mathbf{U} + N_0/(2T)\mathbf{R}_{\mathrm{red}}^{-1}$ as $\mathbf{J}_{\mathrm{red}}^{\mathrm{T}}\mathbf{J}_{\mathrm{red}}$ where $\mathbf{J}_{\mathrm{red}}$ is lower triangular. Noting that the first column of $\mathbf{U}$ is all zeros save its first element which is unity, and the fact that $\hat{\mathbf{W}}_{2:M}$ has only its 1,1 element being non-zero, we see that $\mathbf{U}^{\mathrm{T}}\hat{\mathbf{W}}_{2:M}\mathbf{U} = \hat{\mathbf{W}}_{2:M}$. We can now follow the same reasoning that led to (17). when $\mathbf{R}$ was non-singular, since here we have $\mathbf{R}_{\mathrm{red}}$ which is non-singular. □

B Equivalence of the CM-DFR and the MMSE-DFR

Consider the MSE of the output of a linear feedforward filter for user i when the interference of all previous users has been removed. To show equivalence to the CM-DFR (in the sense that they can use the same feedforward and feedback filters) it is sufficient to show equivalence for the first user, since after the interference of users 1 to $i-1$ have been removed then user i looks like a first user. The MSE of the first user is

$$\mathrm{MSE}_1 = E\left\{(\mathbf{f}^{\mathrm{T}}\mathbf{R}\mathbf{b} + \tilde{n}_1 - b_1)^2\right\}, \tag{27}$$

where $\tilde{n}_1$ has a variance of $N_0/(2T)\mathbf{f}^{\mathrm{T}}\mathbf{R}\mathbf{f}$. When $\mathbf{R}$ is invertible then we can let $\mathbf{k} = \mathbf{R}\mathbf{f}$. This change of variables allows one to show that the MSE is minimized by $\mathbf{k} = w_1 \mathrm{col}_1(\mathbf{S}_{\mathrm{mmse}}^{-1})$ where $\mathbf{S}_{\mathrm{mmse}} = \mathbf{W} + N_0/(2T)\mathbf{R}^{-1}$. If $\mathbf{J}_{\mathrm{mmse}}$ is such that $\mathbf{J}_{\mathrm{mmse}}^{\mathrm{T}}\mathbf{J}_{\mathrm{mmse}} =$

$\mathbf{S}_{\text{mmse}}$ and $\mathbf{J}_{\text{mmse}}$ is lower triangular, then the optimizing $\mathbf{k}$ is $\beta \text{col}_1(\mathbf{J}_{\text{mmse}}^{-1})$ where β is a non-zero constant. So it can be stated that

$$\mathbf{f}_{\text{mmse}} = \beta \mathbf{R}^{-1} \text{col}_1(\mathbf{J}_{\text{mmse}}^{-1}) \tag{28}$$

is the best linear feedforward filter if it is desired to minimize the MSE. From the development of the CM-DFR it was found in (16) that the best feedforward filter is

$$\mathbf{f}_{\text{opt}} = \alpha \mathbf{R}^{-1} \text{col}_1(\mathbf{J}^{-1}), \tag{29}$$

where $\mathbf{J}^{\text{T}}\mathbf{J} = \hat{\mathbf{W}}_1 + N_0/(2T)\mathbf{R}^{-1}$ and α is any non-zero constant. Since $\mathbf{W}$ and $\hat{\mathbf{W}}_1$ are identical except for their $1,1$ elements, $\mathbf{J}$ and $\mathbf{J}_{\text{mmse}}$ differ only in their $1,1$ elements. Knowing this it is then easy to verify that the first column of the inverse of $\mathbf{J}$ is linearly dependent on the first column of the inverse of $\mathbf{J}_{\text{mmse}}$. Hence any feedforward filter that maximizes the SIR will also minimize the MSE and vice versa.

For the case when $\mathbf{R}$ is singular, the quadratic form of the MSE can be equivalently expressed by combining linearly dependent users as was done in Appendix A. Once this transformation has been made, we will have a reduced size correlation matrix (given by the $\mathbf{R}_{\text{red}}$ of Lemma 1) that is invertible so that the optimum feedfoward filter can be found. Once again it will be a non-zero multiple of the feedfoward filter that maximizes the SIR.

References

[1] Cover, T. M., Thomas, J. A., *Elements of Information Theory.* New York: Wiley, 1991.

[2] Gallager, R. G., "A Perspective on Multiaccess Channels," *IEEE Trans. Inform. Theory*, vol. IT-31, No. 2, pp. 124-142, March 1985.

[3] Verdú, S., "Capacity Region of Gaussian CDMA channels: The Symbol- Synchronous Case," in *Proc. Twenty-fourth Allerton Conf. Commun., Contr. and Comput.*, Allerton, IL, Oct. 1986, pp.1025-1034.

[4] Parsavand, D., Varanasi, M. K., "RMS Bandwidth Constrained Signature Waveforms that Maximize the Total Capacity of PAM-Synchronous CDMA Channels," *IEEE Trans. Commun. Theory*, vol. COM-44, pp. 65-75, Jan 1996.

[5] Guess, T. R., Varanasi, M. K., "Onion Peeling for CDMA: Symmetric Capacity Under RMS-Bandwidth Constraints," *Proc. Thirtieth Annual Conference on Information Sciences and Systems*, Princeton University, Mar. 1996, pp. 584-589.

[6] Cheng, R. S., Verdú, S., "Capacity of Root-Mean-Square Bandlimited Gaussian Multi-user Channels," *IEEE Trans. Inform. Theory*, vol. IT-37, No. 3, pp. 453-465, May 1991.

[7] Duel-Hallen, A., "Equalizers for Multiple Input/Multiple Output Channels and PAM Systems with Cyclostationary Input Sequences," *IEEE Journal on Selected Areas in Communications*, Vol. 10, No. 3, April 1992, pp. 630-639.

Design and Performance of Coded High-Density Synchronous CDMA Systems

John C. McCanless†
Stephen G. Wilson†
Thomas R. Giallorenzi‡

†University of Virginia, Charlottesville, Virginia
‡Lockheed Martin Communication Systems, Salt Lake City, Utah

keywords: CDMA, signature sequence sets, error-control coding

Abstract — **A high-density synchronous code division multiple access (S-CDMA) system which utilizes a matched filter receiver is presented and analyzed, demonstrating the potential for higher bandwidth efficiency than conventional (orthogonal) S-CDMA. The signature sequence set of a high-density system has cardinality (number of member sequences) exceeding the processing gain (length of each sequence). Channel capacity studies are performed to determine theoretical limits on system throughput, and bit error performances of some practical situations are analyzed. These results suggest that moderate improvements in spectral efficiency relative to conventional S-CDMA are theoretically possible, but may be impractical.**

1 Introduction

In code-division multiple access (CDMA) systems, each user transmits simultaneously over a common channel, and occupies the same frequency spectrum. Further, each is assigned a signature sequence which, after conventional matched-filter processing of the received signal (a superposition of all users' signals), provides separation of the desired user from the remaining users. This separation is achieved via the inherent correlation function of the matched-filter receiver thus, "good" isolation between the users' signals is obtained with sets of signature sequences having small cross correlation.

In synchronous CDMA (S-CDMA) systems, all users' signature sequences begin and end at the same time, and therefore it is possible to find sets of sequences which are all mutually orthogonal (zero cross correlation). However, since a set of sequences of length P spans at most P dimensions, the number N of mutually orthogonal sequences of length P is at most $N = P$. Nevertheless, such sets are most commonly employed in S-CDMA systems, producing what we shall refer to in this paper as "orthogonal S-CDMA."

Recent work by Massey and Mittelholzer [1, 2] has led to simple constructions of binary-valued (± 1) sequence sets for S-CDMA systems which are "high-density" in the sense that for a given number N of sequences in the set, the length of each sequence is $P < N$. In this case, which we shall refer to as "high-density S-CDMA," there are $N > P$ vectors (signatures) spanning P dimensions, and so there must be some nonzero cross correlation between members of the set. However, Massey and Mittleholzer's constructions minimize the variance of this multiuser interference (MUI) [2, 3]. Thus, under the assumption (due to the Central Limit Theorem) that the MUI experienced by each user is Gaussian, the

constructions are optimal with respect to bit error performance.

2 High-Density Sequence Set Construction

In [2], Massey and Mittelholzer showed that the codewords of a binary linear block code [suitably mapped from GF(2) to ± 1's] constitute an optimal signature sequence set if and only if their *dual* code contains no codewords of Hamming weight two. A linear code satisfies this requirement if its generator matrix has no repeated columns.

We now observe that a Sylvester-type Hadamard matrix of size $N \times N$, $N = 2^k$, is in fact a linear code whose generator matrix consists of all distinct binary k-tuples [4]. For example, the 8×8 Hadamard matrix corresponds to a linear code with generator matrix

$$\mathbf{G}_8 = \begin{bmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{bmatrix}. \quad (1)$$

If we puncture any m columns from the Hadamard generator $\mathbf{G}_N$, we obtain another code with N codewords, each of length $P = N - m$. These codewords correspond to an optimal high-density S-CDMA signature set, since each column in $\mathbf{G}_N$ is still unique. The "density" of this "high-density" set is $N/P > 1$, and can be varied simply by changing the number of punctured columns. As we will see below, density is the most important parameter of an optimal code set since it does not matter which m columns are punctured; that is, the S-CDMA bit error performance is the same for all optimal signature sets of a given density.

3 Capacity Studies

We envision an S-CDMA system in which each of N users employs an error-control code of rate r, whose output symbols are passed to a two-dimensional modulator constellation (such as M-ary PSK or QAM). To each arm of the modulator's (baseband) outputs, each user applies a signature sequence of length P chips, clocked at

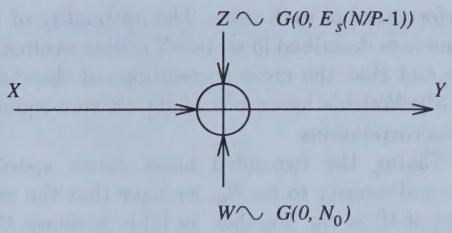


Figure 1: Discrete-time complex channel model. The notation $Z \sim G(\mu, \sigma^2)$ indicates a Gaussian random variable with mean μ and variance σ^2.

P times the modulator's symbol rate, so that the signature sequence repeats once per symbol interval.[1] Finally, carrier waveforms in quadrature are applied to the baseband signal and the result is transmitted to a common additive white Gaussian noise (AWGN) channel, wherein the N signals superpose. Note that symbol timing is strictly synchronized for all the users, and each user is allocated an equal amount of energy per symbol, E_s.

At the receivers, each user performs conventional matched-filter processing (i.e. correlation, or integrate-and-dump). The result is a discrete-time complex channel seen by each of the N users, as diagrammed in Figure 1. It is characterized by the addition of complex white Gaussian noise W and multiuser interference Z, both assumed zero-mean.

Now the MUI is actually a summation of terms which are proportional to the cross correlation of the desired user's signature sequence with that of the remaining users'. Thus the variance of the MUI is proportional to the sum of squares of these cross correlations. Further, since the MUI depends on several interferers, the Central Limit Theorem asserts that the MUI will become approximately Gaussian as the number of interferers grows. Thus, it is only the variances of the MUI and thermal noise which determine the bit error

[1] In [5], a similar model was employed, except that there it was assumed that different signature sequences were to be used in each arm of each user's quadrature modulator. Thus, in that paper a maximum of only $N/2$ users was assumed, since each transmitter employed two of the N possible signatures.

performance of each user. The optimality of the sequences described in section 2 is then assured by the fact that the cross correlations of those sets satisfy Welch's lower bound [6] on sum-squared cross correlations.

Taking the two-sided noise power spectral spectral density to be N_0, we have that the variance of W is N_0. Further, in [3] it is shown that for an optimal signature set of N binary (± 1) sequences $\mathbf{c}_i$, $i \in \{1, ..., N\}$, each of length P, the MUI variance experienced by each user is

$$\mathrm{Var}(Z) = E_s \left(\frac{N}{P} - 1 \right), \tag{2}$$

where E_s is the received energy of one modulator symbol, and is assumed equal for all users.

The critical observation here is that the MUI variance (2) experienced by each user is dependent only on the set's density N/P, and not on the absolute values of N and P. Further, every user experiences this same MUI variance (i.e. the signature set is uniformly good). These properties of optimal code sets imply that we may analyze the performance of high-density S-CDMA for given code set densities without regard to specific constructions of those code sets. [3]

3.1 Dependence of S-CDMA Capacity on E_s/N_0 and N/P

For simplicity, the only modulation we will consider in this paper is M-ary phase shift keying (M-PSK). Recall that the M-PSK signal constellation is two-dimensional for all M, with signal components given by

$$\begin{aligned} X_{i1} &= \sqrt{E_s} \cos\left(\frac{2\pi i}{M}\right) \\ X_{i2} &= \sqrt{E_s} \sin\left(\frac{2\pi i}{M}\right) \end{aligned} \tag{3}$$

where $i \in \{0, ..., M-1\}$.

We then have that, for the high-density S-CDMA channel utilizing an optimal signature set, and under the Gaussian MUI assumption, the channel capacity available to each user is the same, and given by

$$C = \log_2 M - \log_2 \left[2\pi e \left(\sigma_W^2 + \sigma_Z^2 \right) \right] - \frac{1}{M} \int \sum_{i=1}^{M} f(\mathbf{y}|X_i) \log_2 \left(\sum_{i=1}^{M} f(\mathbf{y}|X_i) \right) d\mathbf{y} \quad \text{(bits/use)}. \tag{4}$$

Here we have used the noise and MUI variances in each dimension of the signal space, i.e.

$$\begin{aligned} \sigma_W^2 &= \frac{N_0}{2} \\ \sigma_Z^2 &= \frac{E_s}{2} \left(\frac{N}{P} - 1 \right) \end{aligned} \tag{5}$$

Also, for M-PSK, the conditional pdf's in (4) are, according to our Gaussian assumption,

$$f(\mathbf{y}|X_i) = \frac{1}{2\pi \left(\sigma_W^2 + \sigma_Z^2 \right)} \cdot \exp\left[-\frac{(y_1 - X_{i1})^2 + (y_2 - X_{i2})^2}{2(\sigma_W^2 + \sigma_Z^2)} \right]. \tag{6}$$

To examine the effects of multiuser interference on per-user capacity, Figure 2 plots (4) for increasing density N/P, keeping M constant. The $N/P = 1$ curve, corresponding to the orthogonal S-CDMA case, is included in the plot for each M. Note that for $N/P > 1$, the per-user capacity of the system is strictly less than $\log_2 M$ no matter how high the thermal noise SNR E_s/N_0. Fundamentally, this is because the MUI variance remains constant as the noise variance N_0 is decreased toward zero; therefore, unlike the single-user channel, the multi-user channel cannot be arbitrarily improved simply by increasing E_s/N_0.

Consider now the limiting situation in which the thermal noise energy N_0 is set to zero, and N/P is allowed to vary. Figure 3 plots equation (4) for this situation, versus P/N for convenience. Note that the capacity advantage of 8-PSK over QPSK is quite limited for densities greater than about $N/P = 1.25$; similarly, 16-PSK does not improve upon 8-PSK at all unless the code set density is less than about $N/P = 1.1$.

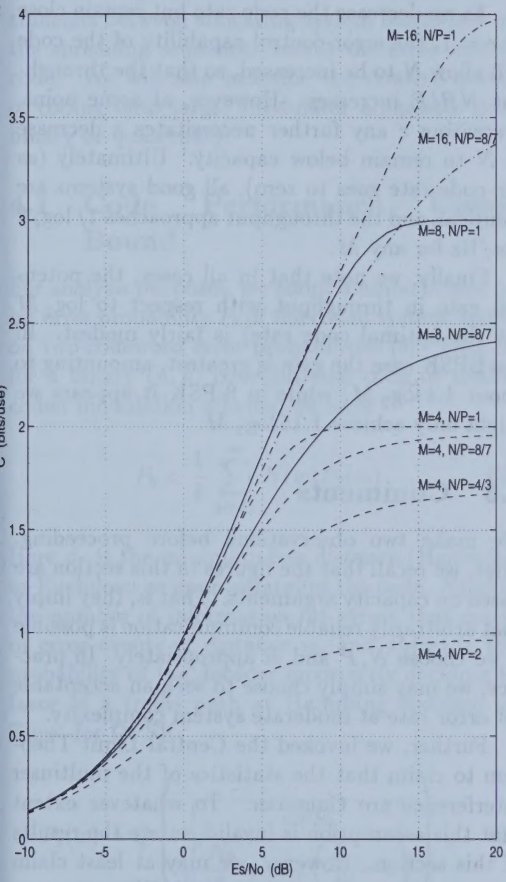


Figure 2: Per-user capacity of M-PSK for several S-CDMA code set densities.

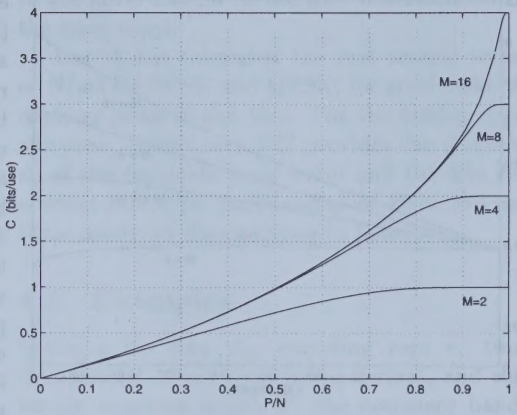


Figure 3: Per-user capacity of M-PSK versus inverse code set density. Thermal noise is assumed zero.

3.2 Spectral Efficiency of M-ary High Density S-CMDA

We now examine the S-CDMA system on the grounds of total spectral efficiency, or throughput, in bits per second per Hertz (bps/Hz). First, we observe that the bandwidth of an M-PSK system operating at a given symbol rate R_s is the same for all M. However, if an error-control code of r is in use, the per-user information rate is $R = r \log_2 M$ bits per M-PSK symbol. Thus the total spectral efficiency η of the S-CDMA system is

$$\eta \triangleq \frac{\text{num. users} \times \text{bits/user/sym.} \times \text{sym. rate}}{\text{Nyquist bandwidth}}$$
$$= \frac{N R R_s}{P R_s}$$
$$= \frac{NR}{P} \text{ bits/sec/Hz}.$$

We now proceed as follows: first, recall that the Noisy Channel Coding Theorem [7] allows us to transmit information arbitrarily reliably at rates less than or equal to capacity, provided we employ an error-control code with rate $R \leq C$. Thus, we choose an information rate $R = r \log_2 M$ and, in Figure 3, we set $R = C$ and locate the corresponding minimum allowable value of P/N on the horizontal axis. This process identifies the maximum allowable throughput $\eta = NR/P$ of

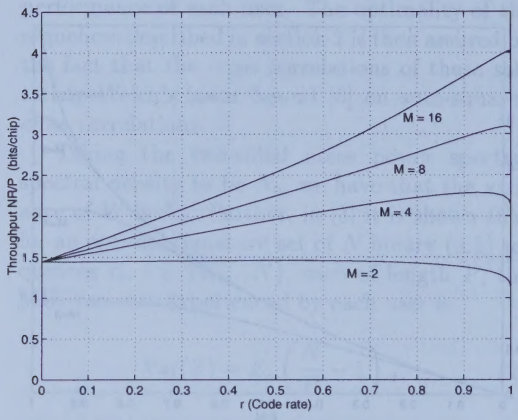


Figure 4: Maximum throughput (bits/chip or bps/Hz) for S-CDMA utilizing M-PSK. Note that thermal noise is taken to be zero.

the S-CDMA system for a given information rate. Repeating the procedure for $R = 0$ to $R = \log_2 M$ yields the data in Figure 4, where for ease of comparison, we have plotted versus the normalized code rate $r = R/\log_2 M$.

Now these curves assume that there is zero thermal noise in the system; were that condition true in an orthogonal S-CDMA system with no coding, the throughput of the system would be $\log_2 M$ bps/Hz. Examining Figure 4 we find that there are regions in each curve where $\eta \geq \log_2 M$ bps/Hz. However, for larger alphabet sizes, the range wherein high-density S-CDMA offers improvement over uncoded orthogonal S-CDMA is increasingly limited.

Worthy of note in Figure 4 is that in each case the throughput of the system falls to $\log_2 M$ bps/Hz for $NR/P = 1$; this situation corresponds to the case of coding with rate $r = 1$ (i.e. no coding), in which case an orthogonal code set ($N/P = 1$) offers the greatest throughput. However, when we apply a code of rate $r < 1$, the error-correcting abilities of the code can (theoretically) compensate for some additional MUI variance; the precise amount is of course a function of the modulation format and code complexity.

Perhaps most interesting here is that there is an optimum value of r which provides the greatest throughput for any M. Our intuition for this (possibly surprising) result is as follows: under a bandwidth constraint, decreasing the code rate r necessitates a decrease in P in order to keep the total spreading P/R constant. The resulting number N of users which can be provided with arbitrarily reliable communication is determined by channel capacity at the given code rate (Figure 3). As we decrease the code rate but remain close to $r = 1$, the error-control capability of the code will allow N to be increased, so that the throughput NR/P increases. However, at some point, decreasing r any further necessitates a decrease in N to remain below capacity. Ultimately (as the code rate goes to zero), all good systems are identical, and the throughput approaches $1/\log_e 2$ bps/Hz for any M.

Finally, we note that in all cases, the potential gain in throughput with respect to $\log_2 M$ (at the optimal code rate) is fairly modest. In the BPSK case the gain is greatest, amounting to about $1.4 \log_2 M$, while in 8-PSK it appears we might only achieve $1.15 \log_2 M$.

3.3 Comments

We make two observations before proceeding. First, we recall that the figures in this section are based on capacity arguments. That is, they imply that arbitrarily reliable communication is possible if we choose N, P and R appropriately. In practice, we may simply choose to seek an acceptable bit error rate at moderate system complexity.

Further, we invoked the Central Limit Theorem to claim that the statistics of the multiuser interference are Gaussian. To whatever extent that this assumption is invalid, so are the results of this section. However, we may at least claim that our assumption of Gaussian MUI is conservative for capacity computations. This follows since for a given variance, Gaussian noise has the greatest entropy, and thus minimizes capacity of the matched-filter channel.

4 Performance of Coded Systems

In this section we examine the performance of some practical S-CDMA systems utilizing optimal signature sets, M-PSK modulation, and finite constraint length convolutional or trellis codes. This investigation is intended to illustrate the tradeoffs between allocating system bandwidth to the spreading sequences versus the error-control code. We will also examine the relative merits of coding onto larger modulator alphabets than binary or quaternary.

4.1 Code Performance Upper Bound

For analysis purposes, we adopt a union bound on bit error probability P_b of the coded system, based on two-codeword error probabilities [8]. That is, for a binary (n, k) convolutional code or trellis-coded modulation scheme, we have

$$P_b < \frac{1}{k} \sum_{w=d_f}^{\infty} N(w) P_2(w) . \tag{7}$$

Here d_f is the minimum free distance (Hamming or Euclidean as appropriate) of the code, $N(w)$ is the number of information bit errors that occur in error events of distance w, and $P_2(w)$ is the probability of the decoder incorrectly deciding in favor of a trellis path at Hamming distance w. Thus, for QPSK,

$$P_2(w) = Q\left(\sqrt{\frac{wrE_b}{\frac{N_0}{2} + rE_b\left(\frac{N}{P} - 1\right)}}\right) , \tag{8}$$

while for higher M-PSK alphabets, where we generally employ trellis-coded modulation,

$$P_2(d) = Q\left(\sqrt{\frac{d^2}{2N_0 + 4E_s\left(\frac{N}{P} - 1\right)}}\right) . \tag{9}$$

In the latter case, d^2 is the Euclidean distance of the error event.

Lee [9] has tabulated the first several terms of $N(w)$ for BPSK and QPSK, for good codes of memory order 8 and less. For the trellis-coded situation, Ungerboeck [10] provides the distance d_f^2 of the dominant error event and the first coefficient $N(d_f)$ for trellis-coded M-PSK. We use these results in the examples to follow.

4.2 Examples

Given a bit rate R_b, encoding rate r, two-dimensional M-ary modulation format, and signature sequence length P, the necessary bandwidth B (i.e. the Nyquist bandwidth) is equal to the chipping rate R_c, so that

$$B = R_c = PR_s = P\frac{R_b}{r \log_2 M} = \frac{PR_b}{R} .$$

The fundamental tradeoff we examine is the following: for a given bit rate R_b and overall bandwidth B, the total allowable spreading, relative to R_b, is fixed at P/R. Thus, we may vary one or more of P, r, or M as desired, but since we agree to keep B and R_b fixed, we must ensure that P/R remains constant.

The following examples help to illustrate the interplay between these system parameters.

Example 1: 64-state codes. We consider two systems, each employing QPSK ($M = 4$) and 64-state convolutional codes. Arbitrarily, we take $N = 128$.

Case 1: $d_f = 4$ We choose a rate $r = 4/5$ code with free distance $d_f = 4$ (a punctured rate 1/2 code given in [9]), and desire to use sequences of length $P = 116$. The effective spreading is then

$$\frac{P}{R} = 116\left(\frac{5}{4 \log_2 4}\right) = 72.5 .$$

The MUI variance for an optimal code set with $N = 128$ and $P = 116$ is

$$E_s\left(\frac{N}{P} - 1\right) = 0.103E_s .$$

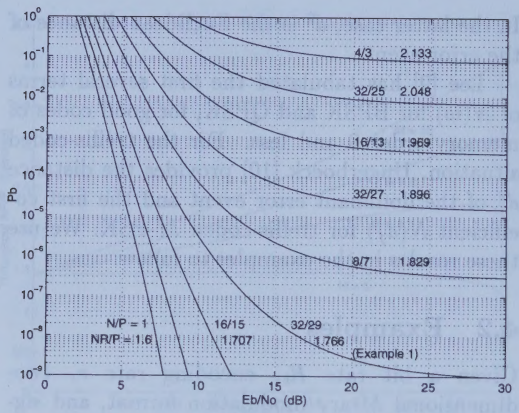


Figure 5: Bit error probability (upper bounds) for high-density S-CDMA using a 64-state, rate 4/5 code along with QPSK. From left to right, successive curves correspond to a decrement of P/N by 1/32. Spectral efficiencies (NR/P) are labelled along with density N/P.

The spectral efficiency of this system is

$$\eta = \frac{NR}{P} = 1.766 \text{ bps/Hz}.$$

In Figure 5 we plot equation (7) for this case, and for several values of code density N/P. Also marked in this figure are the spectral efficiencies NR/P of each system. For the current case, the curve marked "Example 1" applies. Note that this system achieves $P_b = 10^{-6}$ at approximately $E_b/N_0 = 11$ dB, and that the bit error probability has a floor of about $P_b = 10^{-9}$. Also, it is worth noting that the density which achieves $NR/P = 2$ bps/Hz (as in uncoded orthogonal QPSK) has an error floor of about 5×10^{-3}.

Case 2: $d_f = 6$ Suppose now that we are unsatisfied with the bit error probability we achieved in Case 1, and decide to use a code with greater free distance but the same complexity. Then the code rate must be decreased, so we adopt a rate 2/3 code with $d_f = 6$ and continue to use QPSK. In order to maintain the same throughput, then, we must adopt a shorter sequence length P. Specifically, we choose $P = 96$ so that $N/P = 4/3$ and

$$\frac{P}{R} = 96\left(\frac{3}{2\log_2 4}\right) = 72$$

which is approximately the same spreading factor as in Case 1. The throughput (spectral efficiency) for this case is

$$\eta = \frac{NR}{P} = 1.778 \text{ bps/Hz}$$

which of course is also nearly the same as Case 1. However, note that the MUI variance for this system is now

$$E_s\left(\frac{N}{P} - 1\right) = 0.333E_s\,,$$

more than three times that of Case 1.

In Figure 6 we plot equation (7) for Case 2, and once again provide several additional curves corresponding to various code set densities. We immediately observe (on the curve marked "Example 1") that the bit error probability of this system is dramatically higher than that of Case 1. Thus, while the bit error probability of this system is better for any *density* N/P, this system is clearly inferior on an equal *throughput* footing. This example is an illustration of the phenomenon we observed in section 3 (Figure 4), wherein we saw that error-control coding is critical in order to combat MUI, but that lower-rate codes exact a steep price in overall throughput.

Example 2: 256-state codes. Reference to Figures 3 and 4 reveals that in both cases of Example 1, the system is operating below capacity

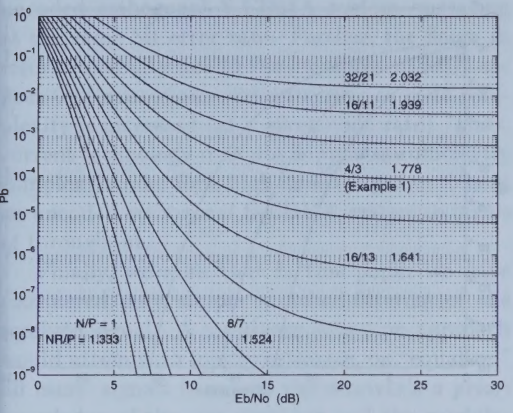


Figure 6: Bit error probability (upper bounds) for high-density S-CDMA using a 64-state, rate 2/3 code along with QPSK.

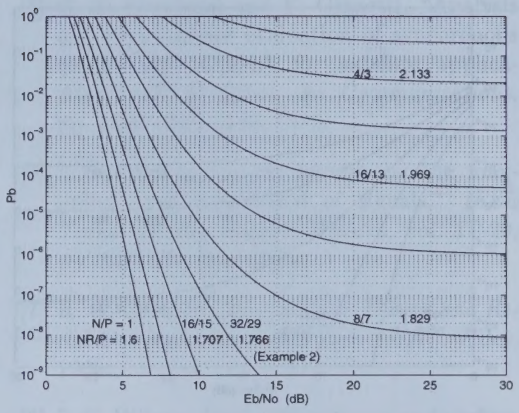


Figure 7: Bit error probability (upper bounds) for high-density S-CDMA using a 256-state, rate 4/5 code (with $d_f = 5$) along with QPSK.

(both in a per-user sense and a system throughput sense). Therefore, the Noisy Channel Coding Theorem asserts that codes exist (perhaps with large complexity) such that the system can operate arbitrarily closely to the capacity predictions of section 3. We therefore examine somewhat more complex convolutional codes, with 256 states. Further, we examine the impact of utilizing a higher alphabet size than QPSK, with the goal of achieving higher throughput.

Case 3: QPSK We adopt the same system parameters as in Case 1, except that we use a memory order 8 (256 state) convolutional code (still of rate $r = 4/5$). QPSK modulation is still assumed, so the system throughput is also the same as Case 1. Figure 7 plots the performance of the system with this more compex code. As the curve marked "Example 2" shows, we achieve $P_b = 10^{-6}$ at approximately $E_b/N_0 = 8.5$ dB, roughly a 2.5 dB improvement over Case 1. Alternatively, note that we could decrement P/N by 1/32 and achieve $P_b = 10^{-6}$ at about 12 dB SNR, which is close to the SNR required in Case 1. The throughput of the system, therefore, could be increased from 1.766 to 1.829 bps/Hz with the latter configuration (relative to Case 1).

Case 4: 8PSK We now seek to achieve higher throughput by utilizing a larger modulator alphabet. We adopt 8-PSK and employ a 256-state, rate $r = 2/3$ trellis code as in [10]. This code has free Euclidean distance of $d_f^2 = 7.515E_s$. A set of performance curves for this architecture is given in Figure 8. Note that for $N/P = 16/15$ we predict $P_b = 10^{-6}$ at about $E_b/N_0 = 11.5$ dB, which is roughly the same energy efficiency as Case 1; however, the real "gain" here is in spectral efficiency, since at this density, the system exhibits a throughput of $\eta = NR/P = 2.13$ bps/Hz.

Example 3: Uncoded Orthogonal S-CDMA.

Case 5: An important comparison to make regarding system throughput is that of high-density coded S-CDMA versus *uncoded orthogonal* S-CDMA. For the latter system, our notation

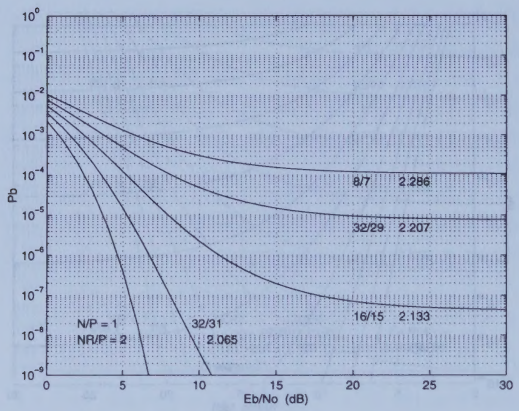


Figure 8: Bit error probability (upper bounds) for high-density S-CDMA using a 256-state, rate 2/3 trellis code (with $d_f^2 = 7.515E_s$) along with 8-PSK.

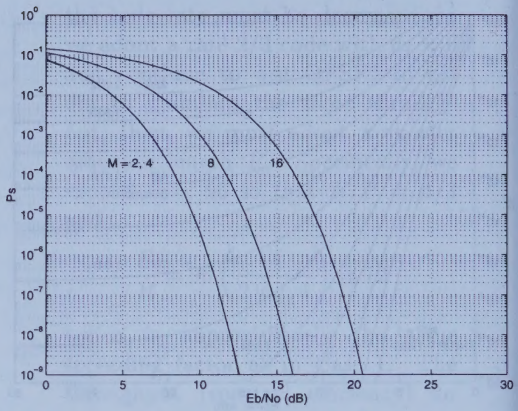


Figure 9: Bit error probability for orthogonal S-CDMA using uncoded M-PSK.

still applies if we take $N/P = 1$ and $R = \log_2 M$. In that case we have $\eta = NR/P = \log_2 M$ bps/Hz.

In Figure 9 we have plotted uncoded M-PSK bit error probabilities versus E_b/N_0. (Here we have calculated exact symbol error probabilities [11], and, assuming Gray-code labeling of the modulator symbols, used $P_b \approx P_s/\log_2 M$.) Note that for 8-PSK at, say, $E_b/N_0 = 14$ dB, we can achieve $P_b = 10^{-6}$ and maintain throughput of $NR/P = 3$ bps/Hz. This is superior performance in both energy efficiency and throughput relative to Case 4. Similarly, at any given E_b/N_0, uncoded QPSK beats the coded high-density S-CDMA systems of Cases 1 through 3.

5 Conclusions

In section 2 we outlined a convenient procedure, derived from the results of Massey and Mittelholzer [2], for the construction of practical, optimal high-density signature sets for S-CDMA.

In section 3 we examined theoretical limits on a high-density S-CDMA system utilizing an optimal code set. We saw that the Noisy Channel Coding Theorem predicted the existence of high-density S-CDMA configurations which can in fact improve upon orthogonal S-CDMA in terms of bandwidth efficiency, while still allowing arbitrarily reliable communication. However, we also saw that this potential exists over a limited range of possible error control code rates, and that in general there is an optimal code rate (near unity) which maximizes the potential throughput of the system. It is intriguing to note that this result is nearly the opposite of that obtained in the asynchronous CDMA case. There, maximum throughput is achieved by very low-rate codes (see [12]), i.e. by allocating bandwidth to the error-control code rather than the spreading sequences. Here, by contrast, we see that one cannot use high-density code sets to "make up" for the bandwidth used by low-rate codes; the MUI variance incurred by doing so is too severe.

In section 4 we quantified this notion further by examining some realistic cases of M-ary phase shift keying used in conjunction with practical convolutional codes. There we found that acceptable bit error rates are achievable for moderate densities (roughly $N/P \leq 1.15$), but that when the error control code rate is accounted for, the overall throughput is often inferior to that of an

uncoded orthogonal S-CDMA system operating at the same bit error rate. Further, the floors in error probability due to the omnipresent MUI appear to decrease only rather slowly as code complexity is increased (at a given code rate). Thus, current technology may not support a sufficiently complex decoder for a high-density system to beat an uncoded orthogonal S-CDMA system in terms of throughput.

Finally, we note that in this paper we have considered only simple matched-filtering of the received signal. A multiuser detection approach would attempt to jointly estimate, or "equalize," all users' signals based on the receiver's *a priori* knowledge of the signature set and the modulation format. Of particular interest here is a recent result due to Rupf and Massey [13], which shows that signature sets which are optimal for the matched-filter receiver architecture are also optimal (in a sum-capacity sense) for the general multiuser channel. However, the authors are not aware of any studies of the performance of multiuser detection schemes employing these sequence sets.

References

[1] J. L. Massey and T. Mittelholzer, "Final report - technical assistance for the CDMA communication system analysis," tech. rep., European Space Agency (ESA-ESTEC) contract 8696/89/NL/US, Institute for Signal and Information Processing, CH-8092 ETH, Zurich, Switzerland, Mar. 1991.

[2] J. L. Massey and T. Mittelholzer, "Welch's bound and sequence sets for code-division multiple-access systems," in *Sequences II*, pp. 63–78, New York: Springer-Verlag, 1993.

[3] J. C. McCanless, "Design and performance of coded high-density synchronous CDMA systems," Master's thesis, University of Virginia, May 1996.

[4] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes.* Amsterdam: North-Holland, 1977.

[5] R. de Gaudenzi and F. Gianetti, "Analysis and performance evaluation of synchronous trellis-coded CDMA for satellite applications," *IEEE Trans. on Communications*, pp. 1400–1408, Apr. 1995.

[6] L. R. Welch, "Lower bounds on the maximum cross correlation of signals," *IEEE Trans. on Information Theory*, vol. IT-20, pp. 397–399, May 1974.

[7] R. G. Gallager, *Information Theory and Reliable Communication.* New York: Wiley, 1968.

[8] S. G. Wilson, *Digital Modulation and Coding.* Upper Saddle River, NJ: Prentice Hall, 1996.

[9] P. J. Lee, "Constructions of rate $(n-1)/n$ punctured convolutional codes with minimum required SNR criterion," *IEEE Trans. on Communications*, vol. COM-36, pp. 1171–1174, Oct. 1988.

[10] G. Ungerboeck, "Trellis-coded modulation with redundant signal sets Part II: State of the art," *IEEE Communications Magazine*, vol. 25, pp. 12–21, Feb. 1987.

[11] R. E. Ziemer and R. L. Peterson, *Introduction to Digital Communication.* New York: Macmillan, 1992.

[12] J. Y. N. Hui, "Throughput analysis for code division multiple accessing of the spread spectrum channel," *IEEE J. Selected Areas Communications*, vol. SAC-2, pp. 482–486, July 1984.

[13] M. Rupf and J. L. Massey, "Optimum sequence multisets for synchronous code-division multiple-access channels," *IEEE Trans. on Information Theory*, pp. 1261–1266, July 1994.

Capacity Analysis of CDMA with Non-equal Cell Loading and Sizes

Seong Won Shin, Soon Man Kwon and Paul. S. Min
Department of Electrical Engineering
Washington University, St. Louis, MO 63130

Abstract

In this paper, we investigate the capacity of the CDMA system when there exists a highly loaded area. In order to provide enough capacity, cell splitting is considered. We choose two scenarios for the highly loaded cells and calculate the capacity of the system with and without an embedded cell. As a generalization, the capacity reduction due to the random characteristics of the cell loading is analyzed. In our numerical results, it is shown that the capacity is strongly affected by the coverage of the embedded cell in both of the scenarios.

1 Introduction

One of the advantages of CDMA systems over conventional cellular systems is the considerable increase of the network capacity. Though CDMA systems provide more capacity than the conventional networks, unusal teletraffic hot spots in a certain cell will easily make the cell site congested and be over the capacity limit. Unfortunately, the congestion due to teletraffic hot spots generates more severe problems in CDMA systems than in frequency reuse-based systems, because adjacent cells will experience the severe interference from these hot spots. The capacity calculation of CDMA systems, however, is not easy due to the fact that unusal behaviors(like high loading and the tendency of users' locations in a cell) of a certain cell affects other cells' capacity severely. In [3], [5], the capacity calculation is done clearly using the same cell interference and other cell interference. But the assumption of uniform loading gives optimistic upper bound for the total capacity of the network, and hence does not explain the effect of unequal cell loading. Let us consider two adjacent cells, 1 and 2. Suppose the number of active users in cell 1 is constantly more than cell 2's. If there is no dynamic change in the cells' coverages, the capacity of cell 1 becomes larger than cell 2's at the expense of the decrease in cell 2's capacity. Hence the sum of two cells' capacity may become less than in the case of equal loading. Moreover, the traffic hot spots affect not only the cells which

contains the hot spots, but also the adjacent cells. The nonuniform distribution of the users over the network, therefore, may result in the severe capacity change.

In [1][4], the embedded microcells were shown to provide solutions to the congested area. Though many of the engineering issues were mentioned regarding microcells, the analysis for the capacity, which is the most important part of embedded microcells, was too simple to be generalized. The realistic environment like soft handoff was not considered. The possibility of overlay structure in CDMA system was also investigated in [2], and their results show that CDMA may not be suitable for the overlaid structure due to the severe interference to the microcells overlaid by a macrocell. The interference calculation was also too simplified. In both of the previous analysis, the suitable capacity comparison between the capacities before and after the adoption of microcells is not done. Moreover, two results in the papers mentioned look somewhat contradictory.

We use the more refined analysis method proposed in [3] for the embedded microcells and additional parameters are considered. In this paper, we investigate the capacity change with the existence of the traffic hot spot. Our analysis considers several scenarios for the traffic hot spots. As a remedy to this problem, cell splitting(an embedded microcell) is considered. In our analysis, the return link capacity is of the main concern, because it is the critical factor for the overall network capacity. In addition, we introduce more reasonable concept of Erlang capacity which is determined by the quality of service.

The remainder of the paper is organized as follows. In section 2, the interference calculation is given, and in section 3, we mention the capacity of the return link. The capacity change with non-uniform cell loading is mentioned in section 4 and the numerical examples are given in section 5. Finally, section 6 provides the conclusions.

2 Interference

As in [3], [5], the propagation loss is modeled as the product of the mth power of distance which is deterministic and a log-normal random component representing shadowing losses. This model represents slowly varying losses, even for users in motion, and applies to both reverse and forward links. Thus, for a user at a distance r from a base station, loss can be represented as

$$L(r, \zeta) = r^m 10^{\zeta/10} \tag{1}$$

where ζ is the decibel attenuation due to shadowing, with zero mean and standard deviation σ. The decibel losses to two or more base stations are assumed to be jointly Gaussian. Hence the random component of the decibel loss to ith base station can be expressed as the sum of two components as

$$\zeta_i = a\xi + b\xi_i \text{ with } a^2 + b^2 = 1, a \leq 1 \tag{2}$$

where ξ is the loss common to all base stations, and ξ_i is the one that is unique to the receiving base station and is independent from one base station to another. The means and variances of ξ and ξ_i are assumed to be identical with 0 and σ^2 respectively.

Hard Handoff(Single Cell Reception) : Single cell reception is a conventional way of controlling a mobile user, i.e. only a single cell's pilot is being tracked at any one time, and hence, the power control is governed by one base station. For simplicity, We assume that the handoff takes place only at the hexagonal boundaries of cells and

users are uniformly distributed over the area of a cell. The user densities for different cells may not be equal. Without loss of generalization, we normalize each cell's radius to unity. Let χ_i be the average density of all active users per unit area in cell i. And also let the distance from the user at certain position to cell i's base station be r_i, and that to cell j's base station be r_j. Since the user is communicating through the nearest base station, it will also be power-controlled by the same base station. Consequently, the relative average interference at cell i's base station due to all users in cell j is

$$\begin{aligned} I_j^i &= E[\iint \frac{L(r_j,\zeta_j)}{L(r_i,\zeta_i)}\chi_j \, dSj] \\ &= e^{b^2(\beta\sigma)^2} \iint R_{ij}^m \chi_j \, dSj \end{aligned} \tag{3}$$

where

$$R_{ij} = \frac{r_j}{r_i}, \quad \beta = ln(10)/10, \text{ and } E[e^{\beta(\zeta_j-\zeta_i)}] = e^{b^2(\beta\sigma)^2}$$

And if the area of a cell is divided into several areas depending upon the users density, then we get

$$\begin{aligned} I_j^i &= \sum_s I_j^i(s) \\ &= e^{b^2(\beta\sigma)^2} \sum_s \iint R_{ij}^m \chi_j(s) \, dS_j(s) \end{aligned} \tag{4}$$

where $S_j(s)$ is a sth partition of the area S_j.
Hence the total interference from other cells to the ith cell is given by

$$\begin{aligned} I_i &= \sum_{j=1,j\neq i}^{N_{Cell}} I_j^i \\ &= \sum_{j=1,j\neq i}^{N_{Cell}} e^{b^2(\beta\sigma)^2} \sum_s \iint R_{ij}^m \chi_j(s) \, dS_j(s) \end{aligned} \tag{5}$$

Consequently, from Eq.(5), we obtain for the mean other-cell interference normalized by N_i,the total number of active users in cell i.

$$\begin{aligned} F_i = \frac{I_i}{N_i} &= (1/(S_i\chi_i))e^{b^2(\beta\sigma)^2} \sum_{j=1,j\neq i}^{N_{Cell}} \sum_s \iint R_{ij}^m \chi_j(s) \, dS_j(s) \\ &= (1/S_i)e^{b^2(\beta\sigma)^2} \sum_{j=1,j\neq i}^{N_{Cell}} \sum_s \frac{\chi_j}{\chi_i} \iint R_{ij}^m \, dS_j \end{aligned} \tag{6}$$

Soft Handoff by Two Cells: By allowing multiple cells to receive active mobile's signals simultaneously, better performance can be obtained than in the case of single cell reception. In soft handoff, the user is connected to two or more cell base stations,

and instantaneously, on a frame by frame basis, the better frame received by either base station is accepted by the network. If a mobile is in a soft handoff region and the forward and the reverse links are balanced, then the mobile is power controlled by the base station which receives the signal of the mobile in the best quality among other base stations which are involved in the soft handoff for the mobile. If the links are not balanced, the quality of the forward link reception by the mobile also should be considered for soft handoff. Suppose mobiles experience a soft handoff by two nearest base stations. Then the mean total interference from the mobiles in the region containing reference base station i, to the ith base station is given as

$$I_{ij}^{i} = E_A[\iint \frac{L(r_j,\zeta_j)}{L(r_i,\zeta_i)}\chi_j\, dS_j] \tag{7}$$

where the expectation is over the acceptable region in the sample space.

$$A = \{(\zeta_i,\zeta_j); L(r_j,\zeta_j) < L(r_i,\zeta_i)\} \tag{8}$$

Let

$$M_j = 10m log_{10} r_j, \text{ and } Q(y) = \int_y^\infty \frac{1}{\sqrt{2\pi}} e^{-x^2/2} dx$$

Then Eq.(7) becomes

$$I_{ij}^{i} = e^{b^2(\beta\sigma)^2} \iint R_{ij}^{m} Q\left(\sqrt{2}b\beta\sigma + \frac{M_j - M_i}{\sqrt{2}b\sigma}\right)\chi_j\, dS_j \tag{9}$$

The interference from the regions in which the reference base station is not involved in a soft handoff($i \neq j,k$) is given as

$$\begin{aligned} I_{jk}^{i} &= e^{b^2(\beta\sigma)^2} \iint R_{ij}^{m} Q\left(b\beta\sigma/\sqrt{2} + \frac{M_j - M_k}{\sqrt{2}b\sigma}\right)\chi_j\, dS_j \\ &\quad + e^{b^2(\beta\sigma)^2} \iint R_{ik}^{m} Q\left(b\beta\sigma/\sqrt{2} + \frac{M_k - M_j}{\sqrt{2}b\sigma}\right)\chi_k\, dS_k \end{aligned} \tag{10}$$

Hence the total interference to the ith base station is obtained by summing all over the regions as

$$I_i = \sum_j \sum_{k,k\neq j} I_{jk}^{i}, \quad j,k = 1,2,\cdots,N_{Cell} \tag{11}$$

Again, the normalized interference can be obtained using Eq.(6).

3 Capacity

Since the CDMA system is interference limited, the performance of the system is characterized by the interference which is experienced by an active user. Let us assume that

there are N_i active users in cell i, where all the users are perfectly power controlled. The activity factor ,ν_k, is a binary random variable indicating whether or not the kth user is active at any instant and let

$$p \triangleq Pr[\nu_k = 1] = 1 - Pr[\nu_k = 0] \tag{12}$$

The total normalized interference, Z_i, at base station i is limited by the interference requirement of a system. If the threshold value for the interference is represented as K'_o, we have

$$Z_i \triangleq \sum_{k=1}^{N_i(1+F_i)} \nu_k \leq K'_o \tag{13}$$

Since N_i is a Poisson random variable with the parameter λ_i/μ, Z_i is also Poisson distributed with the parameter $p\lambda_i(1+F_i)/\mu$.

Now let

$$\rho_i = p\lambda_i(1+F_i)/\mu \tag{14}$$

By approximating the Poisson by a Gaussian with the same mean and variance for large K'_o, the outage probability becomes

$$\begin{aligned} P_o^i &= Pr\left[\sum_{k=2}^{N_i(1+F_i)} \nu_k > K'_o\right] < Pr\left[Z_i > K'_o\right] < e^{-\rho_i} \sum_{k=K'_o}^{\infty} \rho_i^k/k! \\ &\simeq Q\left(\frac{K'_0 - \rho_i}{\sqrt{\rho_i}}\right) \end{aligned} \tag{15}$$

Every network has its quality of service(QOS) requirements, hence it is reasonable that the capacity of a network is to be limited by this QOS requirement. And one of the key factors of the QOS requirements in CDMA is the outage probability, i.e. there exists a certain operating limit in the outage probability to ensure the predefined QOS. Let this limit be P_o^T. Let the maximum operating point of cell i is defined by the maximum number of users in cell i which ensures the outage probability less than P_o^T. Then the maximum operating point of the network or the capacity is obtained as the following.

$$\underset{N_1, N_2, \cdots, N_{Cell}}{maximize} \quad \sum_{i=1}^{N_{Cell}} Ni \tag{16}$$

subject to

$$P_o^i \leq P_o^T, \text{ and}$$
$$\frac{N_i}{N_j} = \frac{\chi_i S_i}{\chi_j S_j} \text{ for } i, j = 1, 2, \cdots, N_{Cell}$$

4 Random Cell Loading

In general, the number of active users(or the density of users) in each cell may not be the same. If the user density is same for all cells all the time, we get the maximum capacity. But fluctuation of user densities in other cells may result in the capacity reduction in

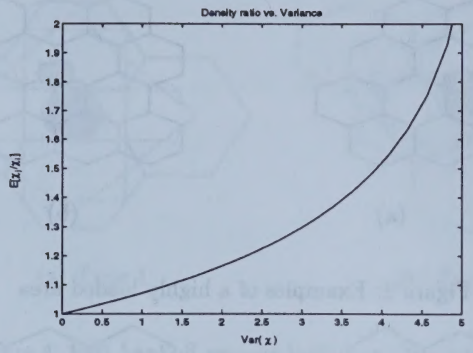


Figure 1: Mean user density ratio

certain cells. Hence it is reasonable that the user density in cells should be random to get the realistic capacities for the cells. Inside cells the density is uniform over a cell coverage region. In a formal description, the number of users in cell i ,N_i (χ_i as a result), is doubly random. As described in the previous section, N_i is a Poisson random variable with the parameter λ_i/μ, where λ_i is ,again, a random variable with a certain distribution.

Since the traffic intensity for each cell of the same type is assumed to be the same(Otherwise, the size of cells should be different), we need to assume that

$$E[\lambda_i] = E[\lambda_j] \text{ or } E[\chi_i] = E[\chi_j] \text{ for } i, j = 1, 2, \cdots, N_{Cell}$$

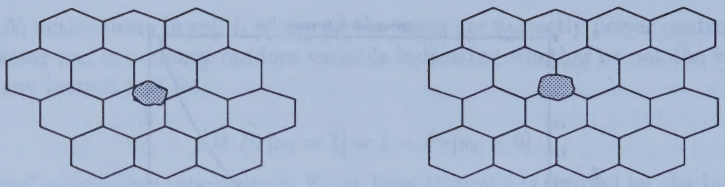
Then Eq.(6) becomes

$$\begin{aligned} F_i^{\chi} &= E_{\chi}[F_i] \\ &= E_{\chi}[(1/(S_i\chi_i))e^{b^2(\beta\sigma)^2} \sum_{j=1, j\neq i}^{N_{Cell}} \iint R_{ij}^m \chi_j \, dS_j] \qquad (17) \\ &= e^{b^2(\beta\sigma)^2}/S_i \sum_{j=1, j\neq i}^{N_{Cell}} \iint R_{ij}^m \, dS_j \left(\iint \frac{x_j}{x_i} f_{\chi_i,\chi_j}(x_i, x_j) dx_i dx_j \right) \qquad (18) \end{aligned}$$

Since χ_i 's are densities of users, we can assume that they are continuous random variables. And χ_i's may not be independent. In Fig. 1, we show the effect of random characteristics of the user densities, where each density χ_i is i.i.d. with a uniform distribution with mean 4. The resultant interference may be more than two times($= E[\frac{\chi_j}{\chi_i}]$) larger at variance 5($= var(\chi_i)$) than in the case of equal deterministic densities. This surely will result in the reduction of the capacity, and becomes more realistic capacity of the network.

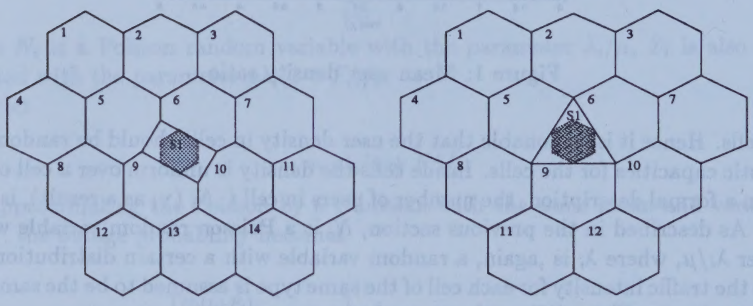
5 Numerical Examples

The service area is covered by equal sized hexagonal cells with normalized radius of 1. The mean active user densities are assumed to be equal. Now we have a teletraffic hot spot which generates several times more load constantly than the one projected. Fig.



(a) (b)

Figure 2: Examples of a highly loaded area

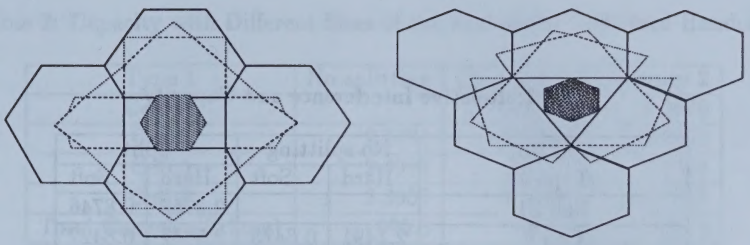


(a) An embedded cell of Type 1 (b) An embedded cell of Type 2

Figure 3: Examples of cell splitting

2 shows two scenarios for the traffic hot spot. Let us assume that the hot spot area is approximated by a hexagonal with a half size of the original hexagonal as shaded regions in Fig.3 and the user density in this area(shaded region) is 3 times larger than the normal density. For the rest of the areas, the densities are the same. Cell splitting is done as shown in Fig.3, where the embedded cell uses the same type of antenna(omnidirectional) with the same power as used in other base stations. For both cases of Type 1 and 2, the coverage areas are, then, obtained with equi-distance lines from the adjacent cells. With the adjustment of antenna height and power, the coverage of the embedded cell can be changed whereas the shape remains the same if the same type of antenna is used. The resulting coverage shapes look strange, but it is due to the adjacent base stations of the same frequency. This is different from the reuse-based systems, where the shape of the coverage of an embedded cell is approximately circular and not affected by the adjacent cells. The soft handoff region in which the small cell is involved is shown in Fig.4. The regions of the soft handoff by two or three base stations including the small cell are shown with dashed lines and dotted lines respectively.

The relative interferences for both scenarios are calculated and shown in Table.1. As can be seen, due to the hot spot, the normalized interferences for the cells not including the hot spot get higher than in normal condition. Cell 6(or 9) containing the hot spot has relatively less interference from the adjacent cells, whereas cell 5(or 10) shows opposite effect due to the huge interference from cell 6. Since cell 6 reaches the QOS requirement first, the total network capacity is determined by this cell. Even when there's enough



(a) Type 1 (b) Type 2

Figure 4: Soft handoff regions including the embedded cell

capacity in cell 5 adding one more user to cell 5 makes the outage probability of cell 6 go beyond the QOS limit. With cell splitting, the interference of cell 5(or 10) gets less due to the embedded cell at the expense of the worse characteristic of cell 6(or 10). But cell 6's increase in the interference is compensated by the embedded cell which shows almost the same characteristic of cell 5. Hence the overall capacity around the hot spot increases. With the threshold value for the interference $K_o' = 230.4$, and the outage probability limit of 0.1, the capacity for two tiers from cell S1(including cell S1) is obtained. We can see almost the same characteristics in Type 2 scenario even though the coverage of S1 is different. Since the hot spot is shared by 3 different cells, the effect is somewhat less than in Type 1. With the embedded cell, cell 6 (or 5,9) experiences more interference whereas cell 10(or 2,8,10) experiences less. The overall capacity gain is similar to Type 1 as can be seen in Table 1. The capacity change in each cell is shown in Fig.5, where the cell type is from the distance of the hot spot. (The embedded cell is type 0 and the cells right next to it are type 1 and so on.) Due to the embedded cell, the adjacent cells experience the capacity reduction whereas it is beneficial to the second and upper tier cells. With the soft handoff, this phenomenon smooths out due to the broadened coverage of each cell. In Table 1, we can see that the advantage of the cell spitting becomes huge with the use of soft handoff.

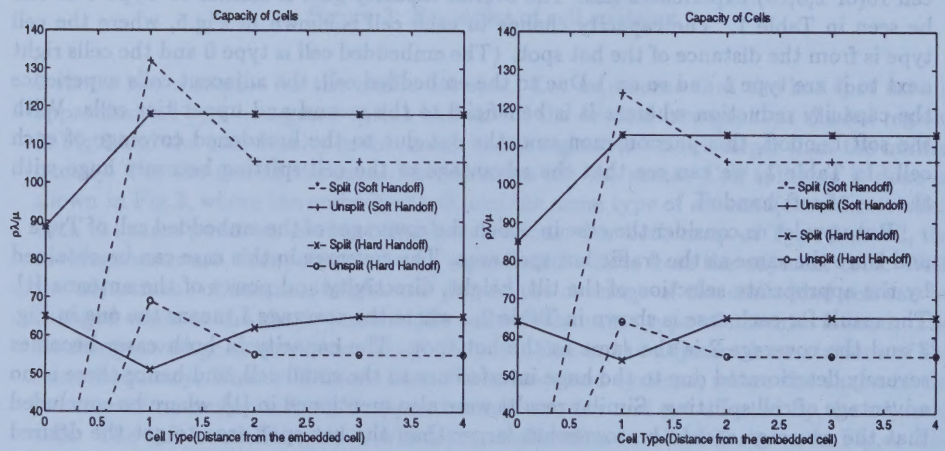
But now let us consider the case in which the coverages of the embedded cell of Type 1 and 2 are the same as the traffic hot spot area. The coverage in this case can be obtained by the appropriate selection of the tilt, height, directivity and power of the antenna [1]. The result for each case is shown in Table 2, , where the coverage 1 means the one in Fig. 3 and the coverage 2 is the same as the hot spot. The capacity in both cases becomes severely deteriorated due to the huge interference to the small cell, and hence there is no advantage of cell splitting. Similar results were also mentioned in [1], where he concluded that the coverage should be somewhat larger than the hot spot area to get the desired capacity. The capacity reduction in this case is mainly due to the adjacent area to the hot spot which no longer belongs to the small cell. Therefore the coverage should be carefully chosen before placing a new cell.

6 Conclusions

We have considered two scenarios for a teletraffic hot spot area. Even though CDMA system is said to adjust easily to the nonuniform cell loading(cell breathing), we showed

Table 1: Relative Interference and Capacity

Cell No.	No splitting		Split	
Type 1	Hard	Soft	Hard	Soft
Cell S1			2.3756	0.8746
Cell 6	2.1491	0.6473	2.5837	0.8422
Cell 5	2.4998	0.8659	2.4196	0.7793
Cell 2	2.3056	0.7644	2.2451	0.7220
Total Capacity(Erlang)	690.5	1320.0	812.5	1504.5
Type 2				
Cell S1			2.3758	0.8471
Cell 5	2.4008	0.7522	2.6664	0.9217
Cell 2	2.3446	0.7843	2.1968	0.7305
Cell 3	2.2946	0.7584	2.2566	0.7365
Total Capacity(Erlang)	685.7	1310.9	812.5	1440.8



(a) Type 1 (b) Type 2

Figure 5: The capacity of the cells

Table 2: Capacity with Different Sizes of the Embedded Cell(Hard Handoff)

Type 1	No splitting	Coverage 1	Coverage 2
Cell S1		2.3756	6.9670
Cell 6	2.1491	2.5837	2.6592
Cell 5	2.4998	2.4196	2.2324
Cell 2	2.3056	2.2451	2.2595
Total Capacity(Erlang)	690.5	812.5	476.0
Type 2			
Cell S1		2.3758	7.8568
Cell 5	2.4008	2.6664	2.3243
Cell 2	2.3446	2.1968	2.2393
Cell 3	2.2946	2.2466	2.2618
Total Capacity(Erlang)	685.7	812.5	425.0

that this advantage no longer applies if the cells adjacent to the highly loaded cell need similar capacity. In our numerical results, we identified that the adjacent cells experience the capacity reduction due to the teletraffic hot spot. The overall capacity with the embedded cell increases with coverage 1, but decreases with coverage 2. Therefore the coverage of an embedded cell should be carefully chosen to contain the area which might generate noticeable interference to the cell if it does not belong to the embedded cell. In addition to the case with the one area highly loaded, we generalized the cell loading and showed the effect of it to the capacity of cells. Due to the random characteristics of cell loading, the capacity of a cell becomes less depending upon the variance of the user densities. Therefore proper capacity anticipation should consider the variation of the cell loading.

References

[1] Joseph Shapira, "Microcell Engineering in CDMA Cellular Networks," *IEEE Trans. on Vehicular Technology* , Vol.43, No. 4, Nov 1994, pp. 817-825

[2] Chih-Lin I, Larry J. Greenstein, and Richard D. Gitlin, "A Microcell/Macrocell Cellular Architectures for Low- and High-Mobility Wireless Users ," *IEEE J-SAC*, Vol.11, No. 6, Aug 1993, pp. 885-891

[3] A. J. Viterbi, A. M. Viterbi, K. S. Gilhousen, and E. Zehavi, "Soft Handoff Extends CDMA Cell Coverage and Increases Reverse Link Capacity," *IEEE J-SAC*, Vol.12, No. 8, Aug 1994, pp. 1281-1288

[4] W. C. Y. Lee, *Mobile Cellular Telecommunications*, McGraw-Hill , 1995.

[5] Andrew J. Viterbi, *CDMA : Principles of Spread Spectrum Communication*, Addison-Wesley , 1995.

Stability of Polling Systems with State-Independent Routing

Serguei Foss*, Natalia Chernova† and Artyom Kovalevskii‡

This paper deals with the stability study of polling systems with either finite or infinite (countable) number of stations (queues) and with a finite number of servers that poll (visit) the stations along some random (state-independent) routes. First, we formulate "global" and "local" stability theorems for systems with a single server and with a general stationary ergodic input. Their proofs are based on certain monotone properties of underlying stochastic processes (see [15-17]) . Second, we give a stability criterion for systems with several servers, with a finite number of stations and with i.i.d. driving sequences. The proof of the latter criterion (see [18]) is based on the fluid approximation approach.

Stability conditions for polling systems have undergone study rather recently (see, e.g., [1-12]) and all available papers deal with systems with finitely many stations and (except [2]) with either Poisson or renewal input.

Keywords: polling system, stability, stationarity, monotonicity, saturation rule, fluid approximation.

§1. Systems with a Single Server: "Global" Stability.

Our approach is based on ideas of the so-called saturation rule [13].

Introduce a polling system with $K \leq \infty$ stations. Let $\langle \Omega, \mathcal{F}, P \rangle$ be a probability space. All random variables below are considered on this space.

The input. By the *input* we mean a marked point process T with points T_n ($T_0 = 0$) and marks ξ_n. The sequence $\xi_n = (\tau_n, \mu_n, \sigma_n), n = 0, \pm 1, \pm 2, \ldots$, is assumed stationary and ergodic. Here $\tau_n = T_n - T_{n-1}$ is the interarrival time between customer $(n-1)$ and customer n, μ_n is the number of the station to which customer n is directed, and σ_n is his service time.

*ColoState University and Novosibirsk State University, foss@stat.colostate.edu and foss@math.nsc.ru

†Novosibirsk State University, root@cher.nsu.ru

‡Novosibirsk State University, artyom@math.nsc.ru

Let $\mathbf{E}\tau_1 = \lambda^{-1}$ be finite and positive; $\mathbf{E}\sigma_1 = \sigma < \infty$; and $\mathbf{P}(\mu_1 = k) = p_k > 0$ for every $k = 1, 2 \ldots$, $\sum_{k=1}^{\infty} p_k = 1$.

The server route. Assume to be given a sequence $\{\nu_j, w_j\}_{j=-\infty}^{\infty}$ of pairs of random variables, where the random variable ν_j takes values $1, \ldots, \infty$ and equals the number of the queue visited by the server jth in succession and the random variable $w_j \geq 0$ is the walking time from queue ν_j to queue ν_{j+1}. Suppose that the sequence $\{\nu_j, w_j\}$ can be partitioned into independent identically distributed (i.i.d.) segments of random length (cycles); i.e., there exists an increasing sequence of integer-valued random variables $\{j_i\}_{i=-\infty}^{\infty}$ such that the random vectors ("cycles of the route")

$$\eta_i = (l_i; \nu_{j_i+1}, \ldots, \nu_{j_{i+1}}; w_{j_i+1}, \ldots, w_{j_{i+1}}), \quad i \in \mathbf{Z},$$

are i.i.d. Here $l_i = j_{i+1} - j_i$ is the number of queues (stations) visited by the server in cycle i. Assume that the cycles start with visiting the same queue. For definiteness, let $\nu_{j_i+1} = 1$ for all i. Denote by $c_k^i = I(\nu_{j_i+1} = k) + \ldots + I(\nu_{j_{i+1}} = k)$ the number of visits to queue k in cycle i, where $\mathbf{P}(c_k^1 > 0) > 0$ for all k, and denote by $\psi_i = w_{j_i+1} + \ldots + w_{j_{i+1}}$ the total walking time during the cycle.

Let $L = \mathbf{E}l_1 < \infty$, $W = \mathbf{E}\psi_1 < \infty$, and $C_k = \mathbf{E}c_k^1 < \infty$ for all k. Assume the sequences $\{\eta_i\}$ and $\{\xi_n\}$ to be independent.

By a *route of the server in the empty system* we mean a marked point random process whose points are the starting times of the cycles and the distance between points equals the total walking time during the corresponding cycle.

Denote by $\Psi = \{\Psi_i, \eta_i\}$ the point process with points Ψ_i and marks η_i, $i \in \mathbf{Z}$, in which $\Psi_0 = 0$ and $\Psi_i = \Psi_{i-1} + \psi_i$ is the finish time of cycle i if the server moves in the empty system.

Consider also a stationary version (in continuous time) of the process Ψ which we denote by $\Psi^{(1)} = (\Psi_i^{(1)}, \eta_i^{(1)})_{i=-\infty}^{\infty}$. Assign the number 0 to the first positive point of this process, so that $\Psi_0^{(1)} > 0 \geq \Psi_{-1}^{(1)}$ a.s.

Denote by $\Psi^{(-n)}$, $n \geq 0$, the stationary ergodic point process that is obtained from the process $\Psi^{(1)}$ by shifting each point to the left by the random variable $\sum_{j=-n}^{0} \sigma_j$ and is renumbered so that $\Psi_0^{(-n)}$ is its first positive point.

Since $\Psi^{(1)}$ and $\{\xi_k\}_{k=-\infty}^{\infty}$ are independent, for every $n \geq -1$ the process $\Psi^{(-n)}$ is independent of the sequence $\{\xi_k\}_{k=-\infty}^{\infty}$ and its distribution coincides with the distribution of $\Psi^{(1)}$.

The service policies. If the server, on visiting station k at time j, finds x customers in a queue, then it serves, without interruption, $f_k^j(x) \equiv f_k(x, D_k^j) \leq x$ customers in the *FIFO* order, and then moves to the next station of the route. Upon service completion, customers leave the system. Here (for every k) the random variables D_k^j, $j = 0, \pm 1, \pm 2, \ldots$, are i.i.d. Suppose that the service policies satisfy the conditions $\mathbf{P}(f_k^j(1) = 1) = \delta_k > 0$ and $f_k^j(x) \leq x$ a.s. for all $x \in \mathbf{Z}^+$, $k = 1, \ldots, K$, and belong to the class $M = \{f : f(x, y) \leq f(x+1, y) \leq f(x, y) + 1$ for all $x \in \mathbf{Z}^+$, $y \in \mathbf{R}\}$. We call the class M the *class of monotone service policies.* For the service policies in M, there always exists a (finite or infinite) limit

$$F_k = \mathbf{E} \lim_{x \to \infty} f_k(x, D_k^j) = \mathbf{E}F_k(D_k^j) \leq \infty.$$

Henceforth the number $[m,n]$ and the arguments Ψ and $T = \{T_k\}$ in some characteristic of the system (queue length, exhaustion time, etc.) signify that the characteristic is considered in the system that is governed by the route Ψ of the server and to which only the customers with numbers $m \le \ldots \le n$ are submitted at respective times $T_m \le \ldots \le T_n$. By a *nonempty* cycle we mean a cycle during which there are customers in the system.

Denote by $X_{[m,n]} = X_{[m,n]}(T, \Psi)$ and $\widetilde{X}_{[-k,l]} = X_{[-k,l]}(T, \Psi^{(-k)})$ the finish times of the last nonempty cycles in the corresponding systems.

Given two systems of the above-described type with (possibly) different arrival times of customers and the service policies, we shall write $T \le T'$ a.s., provided that $T_n \le T'_n$ a.s. for all n, and write $f \ge f'$ a.s., provided that $f_k^j(x) \ge f'^j_k(x)$ a.s. for all $k = 1, 2 \ldots, j = 1, 2, \ldots; x \in \mathbf{Z}^+$.

The above objects enjoy the *monotonicity property*:

If $T \le T'$ and $f \ge f'$ a.s. then

$$X_{[m,n]}(T) \le X_{[m,n]}(T'), \quad \widetilde{X}_{[-k,l]}(T) \le \widetilde{X}_{[-k,l]}(T').$$

Introduce the following notations:

$$Z_{[m,n]}(T) = X_{[m,n]}(T) - T_n, \quad \widetilde{Z}_{[-n,m]}(T) = \widetilde{X}_{[-n,m]}(T) - T_m,$$

$$X_1(T) = X_{[1,1]}(T), \quad Z_1(T) = Z_{[1,1]}(T).$$

Assume the following condition to be satisfied:

(A_1) $\mathbf{E}X_1(T) < \infty$.

The condition (A_1) is always valid for systems with finite number of stations.

Under condition (A_1), we have

Lemma 1 (Law of Large Numbers). There exists a finite constant $\gamma \ge 0$ such that

$$\frac{Z_{[1,n]}}{n} \xrightarrow{\text{P}} \gamma, \quad \lim_{n\to\infty} \frac{\mathbf{E}Z_{[1,n]}}{n} = \gamma;$$

$$\frac{Z_{[-n,-1]}}{n} \xrightarrow{\text{P}} \gamma, \quad \lim_{n\to\infty} \frac{\mathbf{E}Z_{[-n,-1]}}{n} = \gamma.$$

Given an arbitrary $0 \le c < \infty$, denote by cT the process that consists of the points $\{cT_i\}, i \in \mathbf{Z}$, and the marks $(c\tau_i, \mu_i, \sigma_i)$. The monotonicity property and (A_1) imply

Lemma 2 . *For every $c \ge 0$, there exists a nonnegative constant $\gamma(c)$ such that*

$$\frac{Z_{[1,n]}(cT)}{n} \xrightarrow{\text{P}} \gamma(c);$$

moreover, $\gamma(c)$ decreases in c, whereas $\gamma(c) + c\lambda^{-1}$ increases in c.

Denote

$$\gamma(0) = \lim_{c\searrow 0} \gamma(c) = \lim_n \frac{Z_{[1,n]}(0 \cdot T)}{n}.$$

Theorem 1 . *1.* $\gamma(0) = \sigma + \sup_k \frac{p_k}{F_k C_k} W$.

2. There exists $\lim_n \widetilde{X}_{[-n,0]}$ *in the sense of convergence a.s.*

3. The event $\{\lim_n \widetilde{X}_{[-n,0]} < \infty\}$ *has probability 0 or 1.*

4. Let condition (A_1) *hold. If* $\lim \widetilde{X}_{[-n,0]}(T) = \infty$ *a.s. then* $\rho \equiv \lambda\gamma(0) \geq 1$. *If* $\rho > 1$ *then* $\lim \widetilde{X}_{[-n,0]}(T) = \infty$ *a.s.*

For the system governed by the process $\Psi^{(-n)}$, let $Q^k_{[-n,m]}(t)$ stand for the queue length at station k at time t; $\chi^k_{[-n,m]}(t)$, the residual service time at station k at time t; $\chi^0_{[-n,m]}(t)$, the residual interarrival time; $\eta_{[-n,m]}(t)$, the cycle of the route in which the server is at time t (the random vector composed of the numbers of stations and the walking times between them); and $\varphi_{[-n,m]}(t)$, the residual (total) walking time of the server in the cycle $\eta_{[-n,m]}(t)$.

Set the corresponding quantities equal to zero if their values at time t are not defined. All above characteristics are assumed right continuous. Put

$$Y_{[-n,m]} = \{\{Q^k_{[-n,m]}(t)\}_{k=1}^{\infty}, \{\chi^k_{[-n,m]}(t)\}_{k=0}^{\infty}, \eta_{[-n,m]}(t), \varphi_{[-n,m]}(t), 0 \leq t \leq T_m\}.$$

Given random variables X and Y on the probability space $\langle \Omega, \mathcal{F}, P \rangle$, we call X a *copy* of Y if there exits a one-to-one measure-preserving $\mathcal{F}$-measurable shift transformation θ on Ω such that $X(\omega) = Y(\theta\omega)$ for all ω.

We say that the process $\breve{X}(t)$ is *Palm-stationary* (with respect to the nested times $\{T_n\}$, where $T_0 = 0$) if for every n the process $\{\breve{X}^n(t) = \breve{X}(t + T_n),\ t \geq 0\}$ is a copy of the process $\{\breve{X}(t), t \geq 0\}$.

For the process $\{\breve{X}(t),\ \infty \leq t \leq \infty\}$ and for any $m = 1, 2, \ldots$, put $\breve{Y}^m = \{\breve{X}(t), 0 \leq t \leq T_m\}$.

Denote by $Q^k_n = Q^k_{[1,n]}(T_n)$ the queue length at station k at time T_n in the polling system governed by the process Ψ, set $\vec{Q}_n = \{Q^k_n; k = 1, 2, \ldots\}$. Let $Q_n = \sum_{k=1}^{\infty} Q^k_n$ stand for the total queue length.

Theorem 2 . *Assume that condition* (A_1) *holds.*

1. If $\rho < 1$ *then, on the probability space* $\langle \Omega, \mathcal{F}, P \rangle$, *there exists a Palm-stationary process* $\{\breve{X}(t),\ \infty \leq t \leq \infty\}$ *such that for every* $m = 1, 2, \ldots$ *there is a sequence* $\{\breve{Y}^{n,m}\}_{n=1}^{\infty}$ *of copies of the process* $\breve{Y}^m$ *for which*

$$\mathbf{P}(Y_{[-n,m]} \neq \breve{Y}^{n,m}) \to 0 \quad \text{as} \quad n \to \infty.$$

In particular, there exists s stationary sequence $\{\vec{Q}^{(n)}; \infty < n < \infty\}$ *such that*

$$\mathbf{P}(\vec{Q}_n = \vec{Q}^{(n)}) \to 1$$

as $n \to \infty$.

2. If $\rho > 1$ *then there exists* $k < \infty$ *such that* $\sum_{j=1}^{k} Q^j_n \to \infty$ *a.s. as* $n \to \infty$.

Remark 1. It is noteworthy that the existence of the Palm version of a stationary process implies the existence of a stationary process in continuous time (and vice

versa) and some formulas are known that connect the distributions of these marked point processes (see, for instance, [14]).

Remark 2. Extend the class of the service policies under consideration as follows: Let $\widehat{B} = \{f$: for every y there is $\lim_{x\to\infty} f(x,y) = F(y) \leq \infty\}$ and $B = \{f$: for every y there is $\lim_{x\to\infty} f(x,y) = F(y) \leq \infty$, $f(x,y) \leq F(y)$ for all $y \in \mathbf{R}$, $x \in \mathbf{Z}^+\}$.

It is easy to see that $M \subset B \subset \widehat{B}$. For the service policies in the class $\widehat{B}$, there exists

$$F_k = \mathbf{E} \lim_{x\to\infty} f_k(x, D_k^j) = \mathbf{E}F_k(D_k^j) \leq \infty.$$

Theorem 3 *Consider a system with service policies in the class $\widehat{B}$. If $\rho < 1$ then Q_n is bounded in probability; i.e.,*

$$\lim_{x\to\infty} \sup_n \mathbf{P}(Q_n > x) = 0.$$

Theorem 4 *Consider a system with policies in the class B. If $\rho > 1$ then there exists $k < \infty$ such that $\sum_{j=1}^k Q_n^j \xrightarrow{\mathrm{P}} \infty$ as $n \to \infty$.*

The claim of Theorem 4 fails (in general) for systems with policies in the class $\widehat{B}$.

§2. Systems with a Single Server: "Local" Stability .

Consider a model with $K < \infty$ stations. Assume that $\rho > 1$, i.e. the "global" system is unstable. The problem is: do some stable station still exist? We give a positive answer on this question, but under slightly more restrictive assumptions on distributions of driving sequences.

Assume that service time at each station k form a stationary ergodic sequence $\{\sigma_n(k)\}$ with finite mean $\sigma(k)$, all these sequences are mutually independent and do not depend on the sequence $\{(\tau_n, \mu_n)\}$. Put $a_k = \frac{p_k}{F_k C_k} \geq 0$ and permit stations in such an order that

$$a_1 \leq a_2 \leq \ldots \leq a_K.$$

For $k = 1, \ldots, K$, set

$$\rho = \lambda(\sum_{j=1}^{k} \sigma(j)p_j + a_k(W + \sum_{j=k+1}^{K} \sigma_j p_j F_j C_j))$$

(here $0 \times \infty = 0$). Note that $\rho_k \leq \rho_{k+1}$ for all k, and $\rho_K = \rho$.

Theorem 5 *If $\rho_k < 1$ for some $k = 1, \ldots, K$, then there exists a stationary k-dimensional sequence $\{\vec{Q}^{(n)}(k)\}$ such that*

$$\mathbf{P}(Q_n^1, \ldots, Q_n^k) = \vec{Q}^{(n)}(k)) \to 1$$

as $n \to \infty$.

Under assumption (A_1), a similar result takes place for systems with infinite number of stations. One can formulate natural analogs of Theorem 1-4 also.

§3. Systems with Finite Number of Stations and with Several Servers.

Consider a model with $K < \infty$ stations. Assume that interarrival times $\{\tau_n\}$ form an i.i.d. sequence with mean λ^{-1}, each customer (independently of everything else) is sent to station $k = 1, \ldots, K$ with probability p_k. Each server has its own regenerative routing mechanism and service policies. All service policies are assumed to be limited: $F_k^{(m)} < \infty$ for any server m and for any station k. Within one cycle, a server m visits any station k a random number of times with finite mean $C_k^{(m)}$, and its mean cycle walking time is $W^{(m)}$.

Theorem 6 *Assume interarrival times to have an unbounded distribution. Then the model is stable if and only if* $\tilde{\rho} < 1$.

The "unboundedness" assumption may be weakened variously. Here $\tilde{\rho} = \lambda \sum_{r=1}^{R} \beta^{(r)}$, and R and $\beta^{(r)}$, $r = 1, \ldots, R$ are defined by the following recursive procedure.

For $1 \le i, j \le K$, $1 \le m \le M$, put

$$\varphi_{i,j} = \sum_{m=1}^{M} \frac{F_i^{(m)} C_i^{(m)}}{W^{(m)} + \sum_1^j \sigma_k^{(j)} F_k^{(j)} C_k^{(j)}}.$$

Set

$$p_i^{(1)} = p_i; \varphi_1^{(i)} = \varphi_{i,K}; \alpha_i^{(1)} = \frac{p_i^{(1)}}{\varphi_i^{(1)}}; \beta^{(1)} = \min_{1 \le i \le K} \alpha_i^{(1)}.$$

Take $K(1) = K$. Assume that, for some $k \ge 1$, we have previously defined $K(1) > K(2) > \ldots > K(r) \ge 1$ and, for $1 \le k \le r$, $1 \le i \le K(k)$, we know the values of $p_i^{(k)}, \varphi_i^{(k)}, \alpha_i^{(k)}$ and $\beta^{(k)} = \min_i \alpha_i^{(k)}$.

Permit a sequence $\{1, 2, \ldots, K(r)\}$ in such a way that (after permutation)

$$\alpha_1^{(r)} \ge \alpha_2^{(r)} \ge \ldots \ge \alpha_{K(r)}^{(r)}.$$

If $\alpha_1^{(r)} = \alpha_{K(r)}^{(r)}$, then stop with the procedure and put $R = r$. Otherwise, set

$$K(r+1) = \max\{l : \quad \alpha_l^{(r)} > \alpha_{l+1}^{(r)}\}$$

and, for any $1 \le i \le K(r+1)$, put

$$p_i^{(r+1)} = \varphi_i^{(r)}(\alpha_i^{(r)} - \alpha_{K(r)}^{(r)}); \varphi_i^{(r+1)} = \varphi_{i,K(r+1)};$$

$$\alpha_i^{(r+1)} = \frac{p_i^{(r+1)}}{\varphi_i^{(r+1)}}; \beta^{(r+1)} = \min_i \alpha_i^{(r+1)}.$$

References

[1] H. Lévy, M. Sidi, and O. J. Boxma, "Dominance relations in polling systems," Queueing Systems, **6**, 155–171 (1990).

[2] L. Massoulie, "Stability of nonmarkovian polling systems," in: Abstracts: Proc. Conf. Appl. Probab., Paris, 1993. (Submitted to Queueing Systems.)

[3] A. A. Borovkov and R. Schassberger, "Ergodicity of a polling network," Stochastic Process Appl., **50**, 253–262 (1994).

[4] S. Foss and G. Last, "On the stability of greedy polling systems with exhaustive service policies," in: Technical Report 18/94, Technical University Braunschweig, 1994. (To appear in Ann. Appl. Probab.)

[5] S. Foss and G. Last, "On the stability of greedy polling systems with general service policies," in: Technical Report 6/95, Technical University Braunschweig, 1995.

[6] C. Fricker and M. R. Jaibi, "Monotonicity and stability of periodic polling models," Queueing Systems, **15**, 211–238 (1994).

[7] L. Georgiadis and W. Szpankowski, "Stability of token passing rings," Queueing Systems, **11**, 7–34 (1992).

[8] D. P. Kroese and V. Schmidt, "A continuous polling system with general service times," Ann. Appl. Probab., **2**, 906–927 (1992).

[9] D. P. Kroese and V. Schmidt, "Single-server queues with spatially distributed arrivals," Queueing Systems, **17**, 317–345 (1994).

[10] R. Schassberger, "Stability of polling networks with state-dependent server routing," in: Technical Report 12/93, Technical University Braunschweig, 1993.

[11] V. Sharma, "Stability and continuity of polling systems," Queueing Systems, **16**, 115–137 (1994).

[12] H. Takagi, Queueing Analysis of Polling Systems in: Stochastic Analysis of Computer and Communications System, Hideaki Takagi (ed.), Elsevier Science Pub. Horth-Holland (1990).

[13] F. Baccelli and S. Foss, "On the saturation rule for the stability of queues," J. Appl. Probab., **32**, No. 2, 494–507 (1995).

[14] F. Baccelli and P. Bremaud, Elements of Queueing Theory, Springer, Berlin (1994).

[15] S. Foss and N. Chernova, " Comparison theorems and ergodic properties of polling systems," Preprint No. 6, Institute of Mathematics, Novosibirsk (1995) - to appear in Problems of Information Transmission (1996).

[16] S. Foss and N. Chernova, "On polling systems with infinitely many stations," Siberian Math. J., **37**, 940-956 (1996).

[17] S. Foss and N. Chernova, "On a local stability in polling models" (in preparation).

[18] A. Kovalevskii and S. Foss, "Stability of polling models with many servers" (in preparation).

A Graph Based Approach to Reduce Complexity in Queuing Analysis

Gísli Hjálmtýsson*
AT&T Research
Murray Hill, NJ 07974
gisli@research.att.com

Abstract

The simplest queuing systems correspond to birth-death processes, leading to product form solutions. Whereas some other queuing systems are amenable to similar analyses, minor modifications in the problem specification lead to intractable problems. As a result, a large class of queuing problems are frequently solved by simplifying the queuing model until it leads to a product form solution. This talk presents a graph based solution technique, exploiting ideas from graph algorithms and computational complexity, to yield a solution that has a product form over selected paths in the graph.

This technique was developed while working on flow control in ATM networks, specifically in analyzing a buffered moving window schema for input rate control. In this paper, we show how this technique applies to the analysis of a single server system and to multistage service systems obtained by connecting buffered moving windows in tandem. Both the scalar case (where sources are described by an i.i.d. process) and the vector case (where sources are a multiphase process) are considered. The derivations are reduced to graph manipulations and weight reassignments, significantly reducing the total work.

Key words: Queuing analysis, complexity, graph theory, state reduction

1. Introduction

When solving a complex queuing problem, a common approach is to start with a simple case, hoping to learn enough about the problem structure to solve the more general cases. Ideally, the analyses of simple cases should generalize so that the effort is incremental and a function of the differences between the special and the general cases. In queuing analysis, typical scenarios giving rise to such a family of problems come from variations in the governing random processes or the service disciplines [1,2,3,4,5]. Frequently, however, the queuing analysis of each variation reuses little in previous analyses.

Apparently minor variations may dramatically change the complexity of the underlying queuing problem. For example, the simplest queuing systems correspond to birth-death processes, which have simple product-form solutions [1]. Two dimensional birth-death processes defy analysis. Minor modifications, such as service preemption, may change a tractable queuing system to an intractable one.

---

* This research was supported by NSF Grant NCR-9103998, by Minningarsjóður Helgu Jónsdóttur og Sigurliða Kristinssonar, and by Vísindaráð through a Nato Fellowship. The bulk of this work was conducted as part of Ph.D. dissertation work under the supervision of Professor Alan G. Konheim, Computer Science, UC Santa Barbara

Even if generalizations are solvable, analytical approaches frequently exploit problem specifics too early in the analysis to be useful in solving problem variations, requiring completely new analyses even for minor variations. In spite of efforts to analyze more complex queuing systems [1,6,7], large class of queuing problems are frequently approximated by simplifying the queuing models until a product form solution is achievable.

This paper presents a graph based technique for queuing analysis, exploiting ideas from graph algorithms and computational complexity, to yield a solution that has a product form over selected paths in the graph. This approach preserves the problem structure far into the analysis by postponing instance specific assumptions. As a result, the analysis extends to more complex variations. As an application, we analyze a flow control schema for ATM networks, applying this technique starting with the analysis of a single server system, and then generalizing it to a multistage system based on the tandem connection of multiple servers. For each system, both the scalar case where sources are described by an i.i.d. process, and the vector case corresponding to multiphase sources are considered. Having solved the simplest case, we show how incremental effort yields the solutions for the generalizations. The additional derivations are reduced to graph manipulations and weight reassignments.

2. Recursive and Barrier States

Consider an arbitrary queuing problem for which we seek to derive some characteristics. Our goal might be, for example, to derive the queue length distribution or the asymptotic delay. Denote the state space as $\mathcal{V}$, and assume that the proposed solution can be composed from partial solutions conditioned on the states. Investigating each node in the state space, identify each state as a *recursive* or a *barrier* state (residue state). Roughly speaking, the barrier states correspond to a generalized boundary, whereas the recursive states correspond to interior states. Each node is either a recursive state or a barrier state, but not both. This process, therefore, partitions $\mathcal{V}$ into two subsets, which we label red, for the recursive states, and green, for the barrier states. The intent is to express the partial solution for each red state in terms of partial solutions for green states, reducing the problem to analyzing only the partial solutions over the green subset. For this recursion to terminate, we assume that for every red state, either each expansion branch reaches a green, or the problem has a cyclic structure that allows the expansion to be terminated.

As an example consider the simplest queuing problem, a birth-death process, whose state transition diagram is depicted in Figure 1. Following the analysis of [1], each state corresponds to the number of customers in the system, and we seek to derive the probability of having n customers in the system for any n. We identify the zero state as the boundary, which therefore is labeled green, and label all others red. For each red v_i in $\mathcal{V}$, the probability p_i of being in v_i can be expressed in terms of the probability of the (only) green v_0, yielding the familiar formula

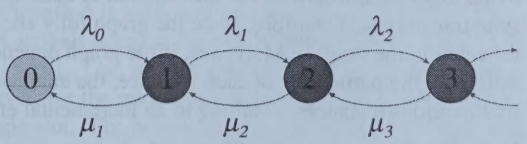


Figure 1 : State transitions for a birth-death process (green - light gray, red - dark gray)

$$p_k = \prod_{i=0}^{k-1} \frac{\lambda_i}{\mu_{i+1}} p_0$$

Reversing the viewpoint, the only green, v_0, occurs in the expansion of every red, with the above coefficient. Consequently, aggregating over (the only) green yields

$$p_0: \quad 1+\frac{\lambda_0}{\mu_1}+\cdots+\prod_{i=0}^{k}\frac{\lambda_i}{\mu_{i+1}}+\cdots=1+\sum_{k\geq 1}\prod_{i=0}^{k-1}\frac{\lambda_i}{\mu_{i+1}}$$

which after normalization fully determines the solution. In general, of course, the set of greens contains more than one state. However, the strategy remains the same, expressing each red in terms of greens and then summing over all greens, applying external conditions to fully determine the final solution.

3. A Graph Based Formulation

In general the approach can be described as an algorithm as follows:

1. **Construct a graph,** $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. The set of vertices, $\mathcal{V}$, corresponds to the state space. The set of edges, $\mathcal{E}$, corresponds to legal state transitions - i.e., $e = (v_i, v_j)$ is in $\mathcal{E}$ if the state transition from v_i to v_j is allowed.
2. **Assign weights to the edges and paths**. The edge weights should account for factors from the backward (or forward) equation(s), that are of value in deriving the solution. The weight of a path is defined as the product of the weight of its edges, i.e.

$$\beta_{\omega,\sigma} = \mathrm{w}(\omega \to \sigma) = \prod_{e \in Path(\omega \to \sigma)} \mathrm{w}(e)$$

3. **Using the path weights**:
   - For each red vertex, recursively expand it in terms of greens.
   - For each green obtain its total contribution, by determining which reds expand to it, and the weights with which it appears in those expansions.
   - Aggregate over the greens.
4. **Finally**, do weight accounting and apply external conditions to determine the solution.

While doing this, the goal is to preserve the structure of the problem as far down the list as possible.

Encoding the problem in a graph, is a complete and very compact representation for Markovian processes, where all the information of consequence can be found in state attributes and state transitions. Therefore, since the graph fully encodes the problem, the solution will be reflected in the graph[†]. Moreover, if the graph depends on the general problem, and varies little with the particulars of each instance, the analysis and the solution extend relatively easily to various instances, resulting in an incremental effort to solve generalizations.

Using this approach, the problem analysis focuses on segmenting the state space by determining the "natural" boundary states, and identifying the paths that reflect the problem structure. We contend that the graph, the boundaries, and the paths of interest are problem attributes, rather than instance attributes, and thus will largely be invariant to minor variations. As a consequence the expansion of red states depends only on problem attributes - the graph, the state partition and potentially a cyclic structure of the problem - and thus *may* also be solely a problem attribute. However, since the weights do depend on the instance, the accounting over greens has to be redone for each instance. Although the total contribution of each green state may have been expressed in path weights, the contribution must eventually

---

[†] Of course a more reflective representation may exist for some problems.

be mapped to concrete representation for algebraic manipulations and simplifications. The solution is completed by applying external conditions.

4. An Application - A Buffered Moving Window

As an application, we analyze an input rate control for ATM networks. Some of the vast amount of work on input rate control and traffic shaping (for example [2,3,8,9,10,11,12]) provides examples of the lack of extendibility of the analysis. In particular, in spite of proposals for multistage entry monitors for policing and shaping according to more detailed traffic descriptors (e.g., mean and peak), the only analytical analysis of such a system is given in [4,5]. In particular the existing analysis of the leaky bucket system does not generalize to more stages.

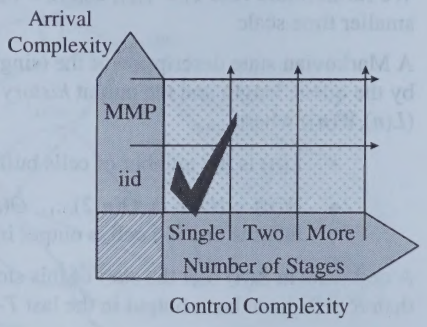


Figure 2 : Extensions in two dimensions

We analyze a buffered moving window entry monitor [13,14]. Its output process is determined by two negotiated parameters (R,T); in every window of size T, no more than R cells are permitted to enter the network. The window moves (slides) in time with each slot so that it always covers the T most recent slots. Cells are buffered to prevent the output rate from exceeding R/T. Starting with an analysis using i.i.d. arrival model and a single stage monitor, the generalizations of interest are in two dimensions: increasing arrival process complexity to model more bursty arrival sources and connecting two or more buffered moving windows in tandem for enhanced control (Figure 2). The aim is to derive a probability generating function for the (combined) queue length distribution of the entry monitor for each of these scenarios. Moreover, using the graph based approach, we seek to analyze the problem using the simplest assumptions, and generalize the analysis to the more complex cases.

4.1 Formulation

The entry monitor is modeled as a discrete-time queuing system. Transmission is over a slotted and synchronous system; the slot length Δ is sufficient to transmit a single cell. We adhere to the ITU-T convention that slot arrivals occur at the end of a slot, whereas transmission begins just after the start of a slot [15,16]. This implies that before a cell can enter the network it *must first be buffered.* However, for the multistage entry monitors we consider, a cell may pass through all stages in a single slot. The arrival of cells is modeled by a sequence of independent and identically distributed (i.i.d.) random variables $\{X(n) : n \geq 1$. $X(n)$ is the number of cells entering in the n^{th}-slot $\Delta(n)$. We use the notation

$$P(z) = \sum_{k \geq 0} p_k z^k,\ pk = \Pr\{X(n) = k\} \quad \mu = E\{X(n)\} \quad \sigma^2 = Var\{X(n)\}$$

for the probability generating function, mean and variance of $X(n)$.

4.1.1 Specification of a Buffered Moving Window

The output process of a single stage buffered moving window is determined by two negotiated parameters (R,T), so that in any window of size T, no more than R cells are permitted to enter the network. The window moves (slides) in time with each slot so that it always covers the T most recent slots. We form multistage entry monitors by connecting number of buff-

ered-moving-windows in tandem. We assume $\mu < R_i/T_i < R_{i+1}/T_{i+1}$; namely the average number of cells entering the buffer in a window of size T is smaller than the maximum number R that can leave the buffer, and that the rates of the stages become progressively larger. We furthermore take $T_i > T_{i+1}$, and $R_i > R_{i+1}$; each stage measures rate on progressively smaller time scale.

A Markovian state description of the (single stage) buffered moving window is determined by the queue length and the output *history* over the last T slots. The state is denoted by $(L(n), W(n))$ where

- $L(n)$ is the number of cells buffered at start of the n^{th}-slot.
- $W(n) = (O(n\text{-}1), O(n\text{-}2), \ldots, O(n\text{-}T))$, is a string of zeros and ones, whose i^{th} element is one if a cell is output in the $(n\text{-}i\text{-}1)^{th}$-slot, and 0 otherwise.

A cell exits in $\Delta(n)$ if at the start of this slot *i)* there is at least one cell buffered, and *ii)* fewer than R cells have been output in the last $T\text{-}1$ slots. The state evolves according to

$$O(n) = 1 \text{ iff } L(n) > 0 \text{ and } \sum_{j=n-(T-1)}^{n-1} O(n) < R$$
$$L(n+1) = L(n) - O(n) + X(n)$$

Although the analysis can be advanced much further using only symbols, define

$$H_\omega(z) = \sum_{i \geq 0} \Pr\{L = i, W = \omega\} z^i$$
$$H(z) = E\left[z^L\right] = \sum_\omega H_\omega(z)$$

Then $H_\omega(z)$ denotes the generating function of the queue length conditioned on the state ω, and $H(z)$ the probability generating function of the (combined) queue length distribution. Using this notation we derive the backward equations

$$\begin{aligned} H_{(0,\tau)}(z) + zH_{(1,\tau)}(z) &= P(z)\left[H_{(\tau,0)}(z) + H_{(\tau,1)}(z)\right] && \text{if } \|\tau\| < R \\ H_{(0,\tau)}(z) &= P(z) H_{(\tau,0)}(z) && \text{if } \|\tau\| = R \end{aligned} \tag{1}$$

4.2 A Graph Based Formulation

Construct a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$. Each vertex, $v \in \mathcal{V}$, is a history of the monitor. The set of edges, $\mathcal{E}$, are the history transitions - i.e., $e = (v_i, v_j)$ is in $\mathcal{E}$ if the transition from the state with history v_i to the state with history v_j is allowed. The transitions of histories are of three types:

a) $(\tau, i) \rightarrow (0, \tau)$ if $\|\tau\| < R$
b) $(\tau, 0) \rightarrow (0, \tau)$ if $\|\tau\| = R$
c) $(\tau, i) \rightarrow (1, \tau)$ if $\|\tau\| < R$

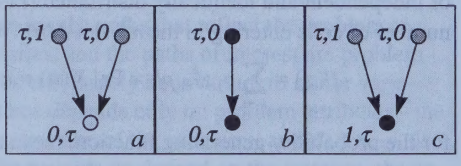


Figure 3 : The three types of transitions

The first transition type corresponds to an idle slot, while the window is not full. However, this implies that at the beginning of the slot, there was nothing to send, i.e. that the queue must have been empty. Consequently, the queue length right after a transition of type *a* is taken is given by the arrival process, which is assumed to be known. We therefore label the vertices $(0, \tau)$ where $\|\tau\| < R$ green. All others are labeled red. We denote the corresponding vertex sets by $\mathcal{V}(G)$, and $\mathcal{V}(R)$ respectively. Moreover, we define $\mathcal{V}(F) \subset \mathcal{V}(R)$, to be the set of histories with exactly R ones. We call

these histories *full*. We note that *exactly* one of the vertices on the left-hand side of the backward Equation 1 is red. The complete graph for a single stage entry monitor is shown in Figure 4.

A *path* ϑ of *length* $m = |\vartheta|$ in $\mathcal{G}$ is a sequence of vertices

$$\vartheta = \langle \sigma, \tau \rangle : \sigma = v_0 \rightarrow v_1 \rightarrow \cdots \rightarrow v_m = \tau$$

where $[v_i, v_{i+1}] \in \mathcal{E}$ is an edge in $\mathcal{G}$, $0 \le i < m$. A *red path*, $\vartheta = \langle \sigma, \tau \rangle$ is one for which all edges are red. Although a red vertex may have two incoming red edges (see Figure 3), any vertex has exactly one outgoing red edge. Thus, by simple induction, if a red path from vertex ω of length k exists, it is unique. Since every red full vertex has an edge to another red full vertex (by cyclic shift), each $\omega \in \mathcal{V}(F)$ is an element of a uniquely determined cycle in $\mathcal{G}: \omega = v_0 \rightarrow v_1 \rightarrow \cdots \rightarrow v_{L-1} \rightarrow v_L = \omega$, where v_{i+1} is the right cyclic-shift of v_i for $0 \le i < L$. The length of the cycle L is a divisor of T, $T = Ls$. We call such a cycle a *max-cycle*. Since a red path from a non-full vertex (red or green) traverses nodes of non-decreasing weight (i.e. non-decreasing number of ones in the history) until it reaches a full vertex, a vertex is at most R edges away from a full vertex.

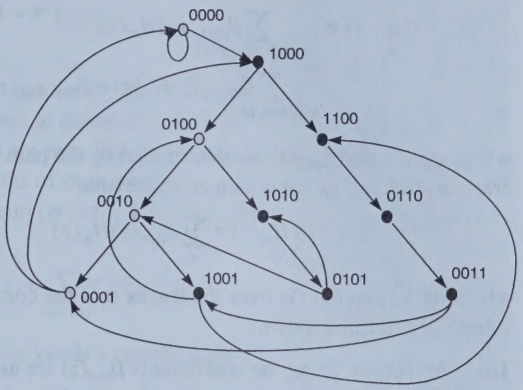


Figure 4 : The complete graph - (T,R) = (4,2)

4.3 Problem Generic Analysis

To determine the paths of consequence in the graph, we study the backward transitions in the graph (i.e. traverse the reverse graph). The strategy is to express reds in terms of greens. From each red history, ω, traverse the graph recursively until either all leaves are green, or the path from ω is of length T. The result is a subgraph rooted at ω. We call this process *backward development*. Backward development corresponds to recursive rearrangement of the backward equations (Equation 1) for a red vertex ω expressing $H_\omega(z)$ in terms of the generating functions of the leafs vertices. In the general case, each backward substitution from the red vertex ω yields one of two possible subgraphs: *i*) a subgraph with 4 vertices if $\omega = (1,\tau)$ where τ contains less than R ones, and *ii*) a subgraph with 2 vertices if $\omega = (0,\tau)$ where where τ contains exactly R ones. The latter subgraph is shown in Figure 3.b, whereas the former equals the combination of Figure 3.a and Figure 3.c.

From the backward equation (Equation 1), we observe that isolating the red vertex $(1,\tau)$ on the left hand side, results in $H_{(0,\tau)}(z)$ having a negative sign. Therefore, backward development in general results in negative and positive terms appearing in the expression for $H_\omega(z)$.

The *weight* of the edge $e = [\sigma, \omega]$ is denoted by $w(e)$; the weight of a path $\vartheta = \langle \sigma, \tau \rangle$ is the product of the weights of each edge in the path, $w(\vartheta) = \prod_{i=0}^{m-1} w([v_i, v_{i+1}])$. By way of demonstration, we use the weight assignments for the single-stage system. Referring once more to the backward equations, Equation 1, we define the edge weights as

$$w(e) = \begin{cases} P(z) & \text{for edge } (\tau, i) \rightarrow (0, \tau) \\ P(z)/z & \text{for edge } (\tau, i) \rightarrow (1, \tau) \end{cases} \quad (2)$$

Figure 5 shows the subgraph determined by $\omega = 0011$, for a single stage system with the parameters $T = 4$, and $R = 2$. The corresponding expression for $H_{0011}(z)$ is given by

$$H_{0011}(z) = \sum_{\sigma\in\{0000,0001,0010\}} \beta_{0011,\sigma}(z)H_{\sigma}(z) - \sum_{\sigma\in\{0000,0001\}} \zeta_{0011,\sigma}(z)H_{\sigma}(z) + \beta_{0011,0011}(z)H_{0011}(z)$$

where $\beta_{\omega,\sigma}(z)$ and $\zeta_{\omega,\sigma}(z)$ are determined by the path weights. In general, for each red ω, we obtain an expression

$$H_{\omega}(z) = \sum_{\sigma} B_{\omega,\sigma}(z)H_{\sigma}(z) \qquad (3)$$

where the summation is over the leaves σ of the complete backward subgraph determined by ω.

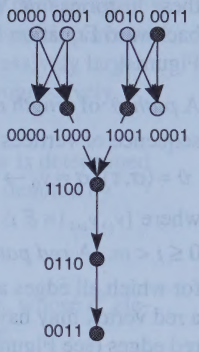


Figure 5 : Example $T = 4$, $R = 2$, $\omega = 0011$

The rules for assigning the coefficients $\beta_{\omega,\sigma}(z)$ are as follows: For a red or positive green leaf σ, the coefficient equals the weight of the path from the leaf to the root ω, i.e. $\beta_{\omega,\sigma}(z) = w(\langle\sigma,\omega\rangle)$. A negative green leaf σ is the end-point of some edge $[\tau,\sigma]$, where τ is either a leaf (green or red) or a red non-full vertex. Since a negative green is the end of two edges, it is helpful to resolve the ambiguity and assume (without loss of generality) that τ ends with a 0. The coefficient of the negative leaf σ is defined as the ratio of the path weight $w(\langle\tau,\omega\rangle)$ to the edge weight $w([\tau,\sigma])$, i.e. $\zeta_{\omega,\sigma}(z) = w(\langle\tau,\omega\rangle)/w([\tau,\sigma])$.

Since we are interested in the summation of $H_{\omega}(z)$ over all ω, Equation 3 defines a term in a summation over all red vertices. Rather than evaluating that double sum, we intend to reverse the order of summation. We must therefore analyze which pairs (σ,ω) appear in the double summation, and particularly *where the vertex $\sigma \in \mathcal{V}$ appears as a leaf.* The most important observation is that only red paths are of interest. Intuitively this means that the solution is determined only by the system behavior while busy. This insight is very valuable, and is the key in understanding later generalizations.

A red vertex σ may appear in subgraphs rooted at several vertices ω; it is a leaf in only the complete backward subgraph rooted at a full vertex and at a distance T from the root (or else growing process could have been continued). Moreover, a red leaf always has a positive coefficient. Since a red path is unique, the root is uniquely determined; i.e. a red vertex appears as a leaf in exactly one complete backward subgraph. Since the path corresponds to a busy period of a full T slots, $\beta_{\sigma,\omega}(z)$ does not depend on σ nor ω; we denote this weight by $\gamma_{\max}(z)$.

A specific green vertex σ is a positive or negative in many complete backward subgraphs. It appears as *positive* in the subgraph of a red vertex ω exactly when there is a path in $\mathcal{G}$ from the green leaf σ to ω of length at most T. For the same reasons as detailed for red leaves, a positive green at distance T from a root ω appears with weight $\gamma_{\max}(z)$. A green vertex σ appears as *negative* in the subgraph of a red vertex ω exactly when there is a vertex τ that is positive in the subgraph of ω and the edge $[\tau,\sigma]$ is in $\mathcal{E}$. Since the length of the path from τ to the root is at most T, and $w([\tau,\sigma]) \neq 1$ a negative coefficient of $\gamma_{\max}(z)$ does not exist.

Figure 6 shows the contribution of the green vertex $\sigma = 0001$. On the left appear the red vertices whose backward development yield σ as a positive leaf, and on the right those that yield σ as a negative leaf. The green vertex on the right is $\tau = 0010$. The coefficients are deter-

mined by the path weights as described above. For example, from Figure 6 we see that σ appears with the coefficient $P(z)/z$ in the expansion of 1000, and with the coefficients

$$\gamma_{\max}(z) = P^4(z)/z^2 \text{ and } -P^3(z)/z^2$$

in the expansion of 0011.

For a green vertex σ let ω_σ be the unique red vertex at distance T from σ, and let ϖ_σ be the unique red vertex at distance T from τ_σ. Note that the path $\langle \sigma, \omega_\sigma \rangle$ and $\langle \sigma, \varpi_\sigma \rangle$ are red. We refer these paths as the positive and the negative paths of σ, respectively. Denote by $pre(\langle \sigma, \omega \rangle, i)$ the prefix of the path $\langle \sigma, \omega \rangle$ of length i, and with $pre(\langle \sigma, \omega \rangle, 0) = \sigma$. Observing that

$$H(z) = \sum_{\omega} H_\omega(z) = \sum_{\omega \in V(G)} H_\omega(z) + \sum_{\omega \in V(R)} H_\omega(z)$$

and using the above, simple algebra finally yields a generic formula in terms of the edge weights, given in

Lemma 1 *Let*

$$\gamma_\sigma(z) = \frac{1}{w([\tau_\sigma, \sigma])} \sum_{i=1}^{R} \left[w\left(pre\left(\langle \sigma, \omega_\sigma \rangle, i-1\right)\right) - w\left(pre\left(\langle \tau_\sigma, \varpi_\sigma \rangle, i\right)\right) \right].$$

Then

$$H(z)(1-\gamma_{\max}(z)) = \sum_{\sigma \in V(G)} \gamma_\sigma(z) H_\sigma(z) \qquad (4)$$

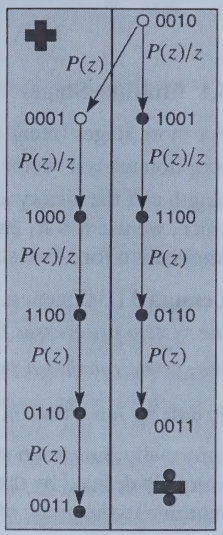


Figure 6 : The contribution of σ = 0001

5. Instance Specific Analysis

To evaluate the formula in Lemma 2 for the various problem instances, weights must be assigned to the edges of the graph. The remaining analysis consists of discovering the weight patterns along the paths of interest, i.e. the positive and negative paths. In particular, since the net contribution of each green vertex is the difference between its positive and negative paths contributions, the path relationship is of primary interest.

For the single stage buffered moving window, the weight assignments are given in Equation 2. With this weight assignment, the weight of a path counts its length as a power of $P(z)$ and counts as a power of $1/z$ the number of transmissions encoded by the path. Since the weight of an edge only depends on the destination, the edge to a negative green always has weight $P(z)$. Furthermore, since $\gamma_{\max}(z)$ corresponds to a busy period of full T slots, and since the system will output *exactly* R cells in those T slots, it follows that $\gamma_{\max}(z) = P(z)^T / z^R$.

In Figure 6 the edges labeled using the single stage buffered moving window wights. The only red vertices that contains 0001 as a negative appear in the right half, whereas those which have 0001 as a positive green leaf appear in the left half. In this case the coefficients of 0001, are

$$\left[\frac{P(z)}{z} + \frac{P^2(z)}{z^2} + \frac{P^3(z)}{z^2} + \frac{P^4(z)}{z^2}\right] - \left[\frac{1}{z} + \frac{P(z)}{z^2} + \frac{P^2(z)}{z^2} + \frac{P^3(z)}{z^2}\right] = \left(1 - \frac{1}{z}\right)\left(1 + \frac{P(z)}{z}\right) + \frac{P^4(z)}{z^2} - 1$$

yielding

$$\gamma_{0001}(z) = \left(1 - \frac{1}{z}\right)\left(1 + \frac{P(z)}{z}\right)$$

In general

$$\gamma_\sigma = \left(1-\frac{1}{z}\right)\left[\sum_{i=0}^{R_\sigma-1}\left(\frac{P(z)}{z}\right)^i\right], \qquad \text{where } R_\sigma \le R$$

5.1 Multiple Stages

For more stages (recall that $\mu < R_i/T_i < R_{i+1}/T_{i+1}$, $T_i > T_{i+1}$, and $R_i > R_{i+1}$) the Markovian description takes a vector form, $(\vec{L},\vec{W})$, where the elements of the vectors describe the queue length and the history at each stage. We use subscripts to identify each stage. For convenience, we use $R = R_1$ and $T = T_1$. Using this notation the conditions on the parameters, and its implication for a two stage system are contained in

Lemma 3 [Uniqueness] *Let $q > 1$ satisfy s.t. $(q-1)R_2 < R \le q\,R_2$, and $qT_2 \le T$. Suppose that the system progresses from the zero state $(\vec{L}_0,\vec{0}), \vec{L}_0 = (0,0)$, to a state $(\vec{L}_{\vec{\sigma}},\vec{\sigma})$. If at any time n, the system reaches a state $(\vec{L}_{\vec{\omega}},\vec{\omega})$ where $\omega_1 = \sigma_1$, then $\omega_2 = \sigma_2$ and $L_{2,\omega} = L_{2,\sigma}$.*

Proof: by induction on n. ■

Informally, Lemma 6 states that under the conditions of the lemma, the two stage history is uniquely defined by the first stage. This implies that the graph for a two stage system is isomorphic to the graph of the single stage system with parameters of the first stage, R and T.

Since the goal for the multistage systems is to describe the combined queue-length distribution, the generating functions remain one dimensional. We define the weights to be

$$w(e) = \begin{cases} P(z) & \text{for edges where the last stage transition is } (\tau,i) \to (0,\tau) \\ P(z)/z & \text{for edges where the last stage transition is } (\tau,i) \to (1,\tau) \end{cases} \qquad (5)$$

This assignment analogous to that of the single stage in that the weight of a path counts its length as a power of $P(z)$ and counts as a power of $1/z$ the transmissions encoded by the path, i.e. number of departures from the last stage. Of course the weight patterns change.

5.2 Multiphase Arrivals

Extending the above for multiphase arrival processes is trivial, the symbols simply represent matrices and vectors appropriately. For example, $\mathbf{H}(z)$, denotes a vector of generating functions, each element corresponding to a phase of the arrival process. Similarly, $\mathbf{P}(z)$ denotes the matrix describing phase transitions and arrival processes of each phase. Observing that all products can be taken from one side (e.g. left), matrix commutativity is not required, and hence the symbolic analysis remains intact.

6. Analytical Results

Using Lemma 1, and the above weight assignments, we obtain the generic analytical result:

Theorem 1: *If $R > T\mu$, (and some additional instance specific conditions)*

$$H(z) = P(z)(z-1)\frac{\sum_{i=0}^{R-1}\alpha_i z^{R-i} P^{\phi(i)}(z)}{z^R - P^T(z)}$$

for some uniquely determined constants $\{\alpha_i\}$ and where $\phi(i)$ depends on the problem instance.

In particular

Lemma 4: *For the single stage Buffered Moving Window, Theorem 1 holds with $\phi(i) = i$.*

Lemma 5: *For the two stage Buffered Moving Window, if* $\exists q : (q\text{-}1)R_2 < R \le q\,R_2$, *and* $qT_2 \le T$ *then Theorem 1 holds with* $\phi(i) = \lfloor i / R_2 \rfloor T_2 + i \bmod R_2$.

For more stages a comparable condition to that of Lemma 6 ensures tractability. However, $\phi(i)$ becomes considerably more complex.

For the vector case the formally equivalent form of Theorem 1 is

$$\mathbf{H}(z)\left(z^R\mathbf{I} - \mathbf{P}^T(z)\right) = \mathbf{P}(z)(z-1)\sum_{i=0}^{R-1}\mathbf{a}_i z^{R-i}\mathbf{P}^{\phi(i)}(z)$$

7. Conclusion

We have shown a graph theoretic framework for queuing analysis. In this framework the problem under study is encoded as a generalized state transition graph. The nodes of the graph encode state information, the edges encode state transitions, and edge weights encode backward equation information. The analysis consists of (1) decomposing the problem into pieces (that may be dependent), each of which has a product form solution, and then (2) combining these partial solutions to form the overall solution. This is achieved by identifying barrier nodes in the graph and expressing all other nodes in terms of these barrier nodes. Analysis of problem variations is reduced to graph manipulations and weight reassignments. Much of the analytic effort is retained in the graph, as extensions are considered.

As an application of these techniques, we presented a discrete time analysis of an input rate control scheme for ATM networks. After solving the simplest problem instance, a single stage system subject to an i.i.d. arrival process, we use graph theoretic arguments and simple weight reassignments to obtain solutions for multistage service disciplines and for multiphase arrival processes.

This work applies to problems that can be decomposed so that each piece in isolation has a product form solution. The problem solution is obtained by aggregating these partial solutions. We conjecture that the set of problems amenable to this approach is significantly larger than the set of problems with product form solutions. The graph theoretic framework unifies and significantly simplifies the analysis.

8. Acknowledgment

Thanks to Albert Greenberg, at AT&T Labs - Research, for his help on the presentation of this work.

References:

[1] L. Kleinrock, *Queueing Systems*, John-Wiley, 1975.

[2] M. Sidi., Wen-Zu, I. Cidon and I. Gopal, "Congestion Control Through Input Rate Regulation," *IEEE Transactions on Communication*, vol. 41, no. 3, 1993, pp. 471-477.

[3] K. Sohraby and M. Sidi., "On the Performance of Bursty Correlated Sources Subject to Leaky Bucket Rate-based Access Control Schemes," in *Proceedings of INFOCOM 1991*, Bar Harbor, FL, April 1991, pp. 426-434.

[4] Gísli Hjálmtýsson and A. G. Konheim, "A Two-Stage Entry Monitor - The Sticky Buffer in Tandem with a Spacer," in *Proceedings of the International Conference on Computer Communications and Networks*, San Francisco, CA, September 1994, pp. 52-58. An extended version to appear in *Telecommunication Systems Journal.*

[5] Gísli Hjálmtýsson, "Input Rate Control and Traffic Shaping in ATM/B-ISDN'," *Ph.D. Dissertation*, University of California, Santa Barbara, June 1995.

[6] N. Götz, H. Hermanns, U. Herzog, V. Mertxiotakis, and M. Rettelbach, "Stochastic Process Algebras - Constructive Specificaton Techniques Integrating Functional, Performance and Dependability Aspects," *Quantitative Modelling in Parallel Systems*, Baccelli and Mitrani editors, Springer, 1995

[7] P. E. Boudreau, J. S. Griffin Jr. and M. Kac, ``An Elementary Queueing Problem," {\em American Mathematical Monthly, October 1962, pp. 713-724.

[8] A. W. Berger "Performance Analysis of a Rate-Control Throttle where Tokens and Jobs Queue," *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 2, February 1991, pp. 165-170.

[9] E. P. Rathgeb, "Modeling and Performance Comparisons of Policing Mechanisms for ATM Networks,"' *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 3, April 1991, pp. 325-334.

[10] P. Boyer, F. M. Guillemin, M. J. Servel, and J-P. Coudreuse, "Spacing Cells Protects and Enhances Utilization of ATM Network Links,"' *IEEE Network Magazine*, vol. 6, no. 5, September 1992, pp. 38-49.

[11] E. Wallmeier, and T. Worster, "The Spacer Policer, an Algorithm for Efficient Peak Bit Rate Control in ATM networks," in *Proceedings of International Switching Symposium*, Yokohama, Japan, October 1992, pp. 451-458

[12] F. M. Brochin, "A Cell Spacing Device for Congestion Control in ATM Networks," *Performance Evaluation*, vol. 16, 1992, pp. 107-127

[13] Gísli Hjálmtýsson and A. G. Konheim, "The Sticky Buffer, A Flow Control Mechanism for ATM," To appear in *Performance Evaluation*.

[14] Gísli Hjálmtýsson and A. G. Konheim, "Analyzing a Two-Stage Entry Monitor for High-Speed Networks," To be presented at *IEEE INFOCOM'97*.

[15] CCITT Recommendation I.371, ``Traffic and Congestion Control in B-ISDN," June 1992.

[16] *ATM : User-Network Interface Specification* (Version 3.0), Prentice-Hall, 1993.

A General Framework for Analyzing Schedulers and Regulators in Integrated Services Networks*

Rajeev Agrawal
Dept. of Electrical and Computer Engr.
University of Wisconsin-Madison
Madison, WI 53706-1691
agrawal@engr.wisc.edu

Rajendran Rajan
IBM T. J. Watson Research Center
P.O.Box 704
Yorktown Heights, NY 10598
raju@watson.ibm.com

Abstract

This paper presents a general framework for obtaining the dynamics of continuous or discrete traffic, and for analyzing end-to-end performance bounds in integrated services networks. These sessions may traverse topologically diverse paths (as with multicast services) and their flow may be moderated by schedulers, regulators, and flow control mechanisms. We first present *synchronous constrained fluid systems* to rigorously obtain the pathwise dynamics of general fluid traffic traversing topologically diverse networks. We then consider a unified characterization of schedulers and traffic regulators for which we obtain end-to-end performance bounds.

1. Introduction

Broadband integrated services networks need to support a wide variety of traffic with diverse characteristics and quality of service (QoS) requirements. Future networks will be designed to preferentially handle real-time applications such as voice or video, providing them QoS guarantees regarding bandwidth, delay, delay jitter, or packet loss. Non-real-time traffic such as data may then be transmitted as "best-effort" utilizing the available bandwidth left over after allocating bandwidth to real-time traffic.

In order to support multiple service categories, networks use scheduling algorithms wherever there is any contention for resources in the network. The primary resource contention is for link bandwidth at the output of switches. At each transmission opportunity, the scheduling algorithm must decide which connection's packet is to be transmitted next. Scheduling algorithms are designed to allow preferential access to bandwidth to higher service categories (by means of a simple priority mechanism, for instance) and to protect individual connections within a service category from one another by guaranteeing each a measure of service independent of the traffic of other connections (by means of mechanisms such as generalized processor sharing (GPS), weighted fair queueing (WFQ), virtual clock (VC), etc.).

In this paper we describe a framework, that of *synchronous constrained fluid systems (SCFS)*, which is used to rigorously obtain the pathwise dynamics of a large class of queueing systems from the topological primitives of the network and the driving arrival and virtual service processes. We allow the driving processes to be discrete, continuous or mixtures of these two, thus representing packetized networks, purely fluid networks or mixtures of the two within the same framework. Further, the networks modelled include tandems, cycles, and fork-join networks with general blocking and starvation, as

*Research supported in part by NSF Grant No. NCR-9305018.

well as networks with splitting and merging of traffic streams. The synchronous timing mechanism used in this paper describes sources and servers with time-varying capacity, as in communication systems with cross traffic (where the server is shared with other traffic streams by means of a scheduling algorithm), or manufacturing systems with time-dependent machine failures.

Following the description of the SCFS model, we obtain structural properties of the synchronous map, and present other equivalent characterizations of this map. Apart from their independent interest, the properties and characterizations play an important role in analyzing the following three problems in the context of scheduling in broadband integrated services networks. First, we consider networks that serve best effort traffic through a strict priority mechanism that utilizes bandwidth leftover from serving higher priority real-time traffic, and seek to balance the costs of increased network provisioning against the benefits of throughput enhancement for the best effort class. The second problem considers the performance of individual real-time connections which have been provided a certain level of isolation from others through scheduling mechanisms such as GPS, WFQ, VC, etc.. The isolation provided by these mechanisms can often be quantified by "service curves", which specify a pathwise service guarantee assured to the connection by the network. Such guarantees may be obtained by placing lower envelope restrictions on the virtual service processes of networks modelled as SCFS, but in fact, can be obtained far more generally. These service curve guarantees are used to obtain performance bounds which in turn can be used to deliver QoS guarantees to such connections. This work can be viewed as an extension of the works of Parekh and Gallager [7],[8], Cruz [4], and Stiliadis and Varma [9]. The last problem we consider is on the use of regulators at various points in the network to smooth or shape individual real-time connections to improve their performance as well as that of other traffic streams at downstream switches. The departure process from a regulator may be characterized in much the same way as an SCFS map with the virtual service process replaced by an upper envelope. Moreover, the departure process so obtained turns out to satisfy the departure process constraints of a service curve scheduler. Thus, regulators may be viewed as special cases of service curve schedulers.

2. The SCFS Model, Properties and Other Equivalent Maps

To illustrate the SCFS model, consider the $\cdot/\cdot/1/K$ queue, which is the fluid analog of the $G/G/1/K$ queue. This consists of a single buffer which stores and releases arriving fluid as in Figure 1. This buffer is fed by server 1 (modelling the arrival process) which

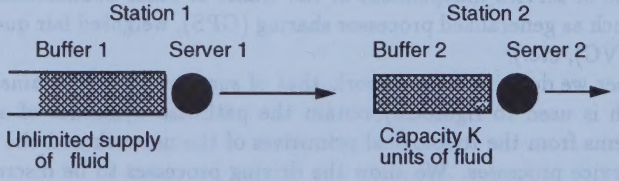


Figure 1. The $\cdot/\cdot/1/K$ Queue.

has an infinite supply of fluid available to it, and is drained by server 2. Fluid may not enter when the buffer is full (*buffer non-overflow*), and cannot be released when the buffer is empty (*buffer non-negativity*). The primitives of this system are the size

of the buffer, denoted by K, and the virtual arrival and service processes, which we represent through the components D_1^* and D_2^* of the *virtual departure process* $\mathbf{D}^*$. In the general fluid model, the virtual departure process $\mathbf{D}^* : \mathbb{R}_+ \to \mathbb{R}_+^2$ is a right-continuous and non-decreasing function, where $D_1^*(t)$ (respectively, $D_2^*(t)$) represents the amount of fluid that would depart from server 1 (resp. server 2), if an infinite supply of fluid were available to that server, and no restrictions were placed on fluid leaving the system. We are interested in the actual departure process $\mathbf{D} = (D_1, D_2) : \mathbb{R}_+ \to \mathbb{R}_+^2$, which describes the amount of fluid that actually departs from the two servers. Roughly, under the synchronous mechanism, we want the "instantaneous rate" of the departure process $\mathbf{D}$ to be synchronized with the "instantaneous rate" of the $\mathbf{D}^*$, and at the same time constrained by the non-overflow and non-negativity constraints on the buffer contents.

For more complicated networks, the construction of the synchronous map would involve cumbersome details of the buffer and initial job configuration, the network topology, routing between stations, etc. We bypass such a network specific construction through the unifying framework of SCFS by transforming network parameters such as configurations of jobs, buffers, the network topology and routing into a *constraint set* $\mathcal{T} \subset \mathbb{R}_+^m$. For the $\cdot/\cdot/1/K$ queue, the constraint set $\mathcal{T} = \{\mathbf{x} \in \mathbb{R}_+^2 : x_1 \leq x_2 + K; x_2 \leq x_1\}$.

2.1 The SCFS Model

Definition 2.1 (Constraint Set). A subset $\mathcal{T} \subset \mathbb{R}_+^m$ is said to be a *constraint set* if it is max-closed, closed, and if every point in $\mathcal{T}$ is accessible from $\mathbf{0}$, i.e., for all $\mathbf{x} \in \mathcal{T}$ there exists a continuous (non-decreasing) path from $\mathbf{0}$ to $\mathbf{x}$ lying entirely within $\mathcal{T}$.

A SCFS is described by a constraint set $\mathcal{T}$, a non-decreasing, right-continuous function $\mathbf{D}^* : \mathbb{R}_+ \to \mathbb{R}_+^m$ called the *virtual departure process*, and a *synchronous map* which specifies how $\mathcal{T}$ and $\mathbf{D}^*$ give rise to the *departure process* $\mathbf{D} : \mathbb{R}_+ \to \mathbb{R}_+^m$.

Definition 2.2 (Synchronous Map $\mathbf{A}$). The *departure process* $\mathbf{D}$ is said to be a solution to the synchronous map $\mathbf{A}$ if it satisfies the following conditions:

A1. $\mathbf{D}(0) \leq \mathbf{D}^*(0)$.

A2. $\mathbf{D}(t) - \mathbf{D}(s) \leq \mathbf{D}^*(t) - \mathbf{D}^*(s)$, for all $0 \leq s \leq t$.

A3. $\mathbf{D}(t) \in \mathcal{T}$ for all $t \geq 0$.

A4. $\mathbf{D}$ is the (pointwise) maximal function satisfying the above conditions.

In that case we write $(\mathcal{T}, \mathbf{D}^*) \xrightarrow{s} \mathbf{D}$.

The first theorem asserts that we may always find a departure process generated by the synchronous mechanism, and that this process is unique.

Theorem 2.3. *Given a constraint set $\mathcal{T}$ and a virtual departure process $\mathbf{D}^*$, there exists a unique function $\mathbf{D} : \mathbb{R}_+ \to \mathbb{R}_+^m$ satisfying* **A1**–**A4** *above.*

2.2 Structural Properties

A number of properties of the synchronous map $(\mathcal{T}, \mathbf{D}^*) \xrightarrow{s} \mathbf{D}$ fall out as a simple consequence of the definition. These include:

1. **Scaling:** Scaling the constraint set and the virtual departure process appropriately, results in a departure process that is scaled likewise.

2. **Monotonicity:** The departure process $\mathbf{D}$ is non-decreasing as a function of the constraint set and the virtual departure process.

3. **Concavity:** The departure process $\mathbf{D}$ is jointly concave as a function of the constraint set $\mathcal{T}$ and the virtual departure process $\mathbf{D}^*$.

2.3 Other Equivalent Synchronous Maps

In this section we present a number of different ways of obtaining the actual departure process $\mathbf{D}$ from the virtual departure process $\mathbf{D}^*$ of a SCFS described by its constraint set $\mathcal{T}$. We first introduce the *constraint function* $\mathbf{M}$ as an alternative way of capturing the topological primitives of the system and show that it is equivalent to specifying the constraint set $\mathcal{T}$. We then introduce the different synchronous maps and show that they give rise to same actual departure process.

Definition 2.4 (Constraint Function). A function $\mathbf{M} : \mathbb{R}_+^m \to \mathbb{R}_+^m$ is said to be a *constraint function* if $\mathbf{M}$ is non-decreasing and continuous from above, and if $M_i(\mathbf{x})$ does not depend on x_i for $i = 1, \ldots, m$,

Definition 2.5. Given any subset $\mathcal{T} \subset \mathbb{R}_+^m$ define the induced function $\mathbf{N}(\mathcal{T}) : \mathbb{R}_+^m \to \mathbb{R}_+^m$ by

$$N_i(\mathcal{T})(\mathbf{x}) := \sup\{y \geq 0 : \exists \mathbf{z} \in \mathcal{T} \ni z_i = y, z_j \leq x_j \forall j \neq i\}, \quad i = 1, \ldots, m.$$

(Note that $N_i(\mathcal{T})(\mathbf{x})$ does not depend on x_i.) Also, given any function $\mathbf{M} : \mathbb{R}_+^m \to \mathbb{R}_+^m$ define the induced set $\mathcal{U}(\mathbf{M}) \subset \mathbb{R}_+^m$ as

$$\mathcal{U}(\mathbf{M}) := \{\mathbf{x} \in \mathbb{R}_+^m : \mathbf{x} \leq \mathbf{M}(\mathbf{x})\}.$$

Theorem 2.6. *If $\mathcal{T}$ is a constraint set then the induced function $\mathbf{N}(\mathcal{T})$ is a constraint function and $\mathcal{U}(\mathbf{N}(\mathcal{T})) = \mathcal{T}$. If $\mathbf{M}$ is a constraint function then the induced set $\mathcal{U}(\mathbf{M})$ is a constraint set and $\mathbf{N}(\mathcal{U}(\mathbf{M})) \leq \mathbf{M}$.*

Henceforth we assume that $\mathcal{T}$ is a constraint set and $\mathbf{M}$ is a constraint function such that $\mathbf{N}(\mathcal{T}) = \mathbf{M}$ and $\mathcal{U}(\mathbf{M}) = \mathcal{T}$. Let $\mathbf{D}^* : \mathbb{R}_+ \to \mathbb{R}_+^m$ be a virtual departure process (i.e., non-decreasing and right-continuous). We now give several different ways in which the actual departure process $\mathbf{D} : \mathbb{R}_+ \to \mathbb{R}_+^m$ may be obtained. These equivalent characterizations are based on reflection maps used by Harrison and Reiman to characterize the dynamics of certain queueing networks [6].

Equivalently we may define the *loss* process $\mathbf{C} : \mathbb{R}_+ \to \mathbb{R}_+^m$ where $\mathbf{C} = \mathbf{D}^* - \mathbf{D}$. Consider the following maps:

Definition 2.7 (Synchronous Map **B**). $\mathbf{D}$ is said to be a solution of the synchronous map $\mathbf{B}$ if it satisfies the conditions **A1**-**A3** and

B4. For each $i = 1, \ldots, m$, C_i increases only when $M_i(\mathbf{D}(t)) - D_i^*(t) + C_i(t) = 0$, i.e.,

$$\int_{[0,\infty)} 1_{\{M_i(\mathbf{D}(t)) - D_i^*(t) + C_i(t) > 0\}} dC_i(t) = 0, \quad i = 1, \ldots, m.$$

Definition 2.8 (Synchronous Map **C**). $\mathbf{D}$ is said to be a solution of the synchronous map $\mathbf{C}$ if it satisfies the following conditions:

C1. $\mathbf{D}$ is non-decreasing.

C2. $\mathbf{D}$ is a fixed point of the following equation:

$$D_i(t) = \inf_{0 \le s \le t} [D_i^*(t) - D_i^*(s) + M_i(\mathbf{D}(s))] \wedge D_i^*(t), \quad t \ge 0, i = 1, \ldots, m.$$

Definition 2.9 (Synchronous Map $\mathbf{D}$). $\mathbf{D}$ is said to be a solution of the synchronous map $\mathbf{D}$ if it satisfies the condition:

D. $\mathbf{D} := \lim_{n \to \infty} \mathbf{D}^n$ where $\mathbf{D}^0 = \mathbf{D}^*$ and $\mathbf{D}^n, n \ge 1$ are defined inductively as follows:

$$D_i^{n+1}(t) = \inf_{0 \le s \le t} [D_i^*(t) - D_i^*(s) + M_i(\mathbf{D}^n(s))] \wedge D_i^*(t), \quad t \ge 0, i = 1, \ldots, m.$$

Theorem 2.10. *There exists a unique solution to each of the synchronous maps* **A**, **B**, **C**, **D**. *These solutions are identical.*

3. Performance of Best Effort Traffic

Consider a non-real time multicast connection with hop-by-hop flow control in a broadband integrated services network. This connection originates at a source node which has an inexhaustible supply of data (an "eager" or "greedy" source), with a number of destination nodes as shown in Figure 2. The source is connected to each of the destination

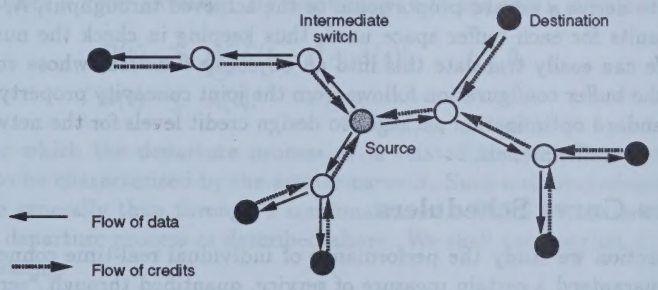


Figure 2. Multicast connections with hop-by-hop credit-based flow control.

nodes through a number of intermediate nodes and links. We assume that the traffic corresponding to this multicast connection is delay insensitive but loss sensitive (bulk e-mail, for instance), and hence the connection is being carried as "best effort" traffic, such as the ABR service category in ATM networks or legacy traffic in a QoS capable IP network. In either case, the capacity that is allocated to the multicast connection is the residual capacity available after serving the higher priority real-time cross-traffic; VBR or CBR in the ATM context. The flow control mechanism in effect is hop-by-hop and credit based, i.e., each node periodically issues credits to its immediately upstream node about the volume of traffic it can receive without any buffer overflow. The upstream node is constrained by credits it has outstanding from each of its immediately downstream nodes, and the number of outstanding credits is diminished on the transmission of data from the upstream node. Consequently, the upstream node can transmit as many packets as the minimum of all the outstanding credits it has.

We may model the multicast connection as a forking network with finite buffers and communication blocking. Let $\mathbf{k} = (k_1, \ldots, k_m)$ denote the capacities of buffers at various stations, where $k_1 = \infty$. In order to model this system as a SCFS, define the matrix $\{u_{ij}\}_{1 \leq i,j \leq m}$, with $u_{ij} = 0$ if station j is immediately upstream of station i, $u_{ij} = k_j$ if it is immediately upstream, and $u_{ij} = \infty$ otherwise. The set $\mathcal{T} = \{\mathbf{x} \in \mathbb{R}_+^m : x_i \leq x_j + u_{ij}; 1 \leq i, j \leq m\}$ may be verified to be a constraint set that captures the topological primitives of the multicast connection.

Let $D_1^* : \mathbb{R}_+ \to \mathbb{R}_+$ denote the cumulative source traffic generation process, and $D_i^* : \mathbb{R}_+ \to \mathbb{R}_+$ the cumulative effective service capacity that is available to the virtual circuit at node i, $2 \leq i \leq m$, after taking into account capacity that has been diverted to service higher priority traffic. Several models have been proposed for the cumulative source traffic generation process D_1^* and the intermediate virtual departure processes $(D_2^*, \ldots, D_m^*)$, in order to capture the features of real traffic on broad-band networks. These include the Markov modulated Poisson model [5], the related fluid model [1] long-range dependent fluid models [2], or (σ, ρ) models which enforce burstiness constraints on the general fluid virtual departure processes [3]. In any event, the actual traffic on the network, described by the departure process $\mathbf{D}$, may be obtained through the map $(\mathcal{T}, \mathbf{D}^*) \xrightarrow{s} \mathbf{D}$.

As an instance of the problems that may be addressed by the results in this paper, suppose that we wish to determine the aggregate number of buffer spaces (credits) that should be dedicated to this connection at all nodes, as well as obtain a particular configuration of buffer sizes $(k_1, \ldots, k_m)$ that should be assigned to individual nodes. The objective is to derive a reward proportional to the achieved throughput, Λ, while paying a cost of c units for each buffer space used, thus keeping in check the number of cells in flight. We can easily translate this into an objective function, whose concavity as a function of the buffer configuration follows from the joint concavity property. This allows us to use standard optimization packages to design credit levels for the network, under a wide variety of traffic models.

4. Service Curve Schedulers

In this section we study the performance of individual real-time connections which have been guaranteed a certain measure of service, quantified through "service curves". Such guarantees may reflect active scheduler involvement in isolating flows (as in WFQ, GPS, etc.), or may be obtained through a careful accounting of the service capacity and arrival constraints on other connections (when several burstiness constrained flows share a FIFO scheduler, for instance). Often times, service curves can result as a combination of a SCFS model with a lower envelope on the virtual departure process. However, the service curve (partial) characterization given below can arise more generally. The principal merit of the this general characterization is that it allows us to capture the relevant service characteristics of scheduler, as well as the interaction effects of cross-traffic, neccessary to calculate worst case queue-length and delay bounds for a connection. Further, service curves may be easily aggregated, resulting in tighter end-to-end bounds than the aggregation of bounds obtained from elements considered in isolation. Service curves were introduced by Parekh and Gallager [7], [8], in the context of Generalized Processor Sharing, and extended by Cruz [4], and Stiliadis and Varma [9]. The service curve characterization introduced later in this section is more general than those found in the works cited before, and includes both discrete as well as continuous traffic models within the same analytic framework. The greater generality allows us to obtain tighter

bounds in certain cases, and results in considerably simplified proofs of various end-to-end performance bounds.

We first present the analysis for a single scheduler and then extend this to a tandem of such schedulers. Consider the synchronous map C defined in Section 2.3 for a $\cdot/\cdot/1/\infty$ queue without blocking. For such a queue the actual arrival process is equal to the virtual arrival process, i.e., $A := D_1 = D_1^*$. Denote the virtual departure process by $D^* := D_2^*$ and the actual departure process by $D := D_2$. Then D is given by

$$D(t) = \inf_{0 \le s \le t} [D^*(t) - D^*(s) + A(s)] \wedge D^*(t), \quad t \ge 0.$$

If we define all virtual and actual departure processes on the entire real line with the assumption that they are identically zero on the negative half, then the above equation is equivalent to:

$$D(t) = \inf_{s \le t} [D^*(t) - D^*(s) + A(s)], \quad t \ge 0. \tag{4.1}$$

Assume that the virtual departure process D^* has a lower envelope S, i.e.,

$$D^*(t) - D^*(s) \ge S(t-s) \quad \forall s \le t. \tag{4.2}$$

As mentioned before such a lower bound on the virtual departure may arise in certain multiplexed systems. For instance, when the service allocated to the traffic in consideration is the residual capacity of a shared server after having served a higher priority cross traffic, and the cross traffic has an upper envelope, then the virtual departure process for this traffic satisfies a lower envelope. By combining (4.1) and (4.2), it follows that

$$D(t) \ge \inf_{s \le t} [S(t-s) + A(s)], \qquad t \ge 0, \tag{4.3}$$

$$D(t) \le A(t), \qquad t \ge 0. \tag{4.4}$$

Any queue for which the departure process D is related to the arrival process A as above is said to be characterized by the *service curve* S. Such a characterization may be obtained more generally than through a combination of a SCFS with a lower envelope on the virtual departure process as described above. We shall assume that $S(0) = 0$ and S in non-decreasing.

Next, we obtain various deterministic bounds for a queue with a service curve characterization. Some of the results presented below are pathwise in the arrival process A, while for others we assume that the arrival process has an upper envelope or a *burstiness constraint*, i.e.,

$$A(t) - A(s) \le B(t-s) \quad \forall s \le t. \tag{4.5}$$

As for S, we shall assume that $B(0) = 0$ and that B is non-decreasing.

Proposition 4.1 (Upper Bound on Queue Size). *For each arrival process A, the queue length Q is bounded above by*

$$Q(t) := (A(t) - D(t)) \le \sup_{s \le t} ((A(t) - A(s)) - S(t-s)), \quad t \ge 0.$$

If the arrival process has a burstiness constraint B, then the queue length is bounded above by

$$Q(t) \le \sup_{0 \le s \le t} (B(s) - S(s)) \vee (B(t^+) - S(t^+)) =: \Sigma(t) \le \sup_{s \ge 0} (B(s) - S(s)) =: \Sigma, \quad t \ge 0.$$

Proposition 4.2 (Upper Bound on Burstiness of Departure Process). *If A satisfies the burstiness constraint B, the departure process D satisfies the burstiness constraint $B+\Sigma$, i.e.,*

$$(D(t) - D(s)) \leq B(t-s) + \Sigma, \quad \forall s \leq t.$$

Next we bound the delay associated with such a queue. Let $\Delta(t)$ be the additional time after t at which all arrivals up to (and including) at time t depart from the queue, i.e.,

$$\Delta(t) := \inf\{\delta \geq 0 : D(t+\delta) \geq A(t)\}.$$

Proposition 4.3 (Upper Bound on Delay).

$$\Delta(t) \leq \inf\{\delta \geq 0 : S(u+\delta) - B(u) \geq 0 \quad \forall u \geq 0\}, \quad t \geq 0.$$

4.1 Open Tandem of Schedulers

Now consider an open tandem of m service curve schedulers. Let the arrival process to the tandem be denoted by A, the departure process by D, the arrival process into scheduler k by A_k and the departure process by D_k, $1 \leq k \leq m$. Then

$$A_1 = A, \quad A_k = D_{k-1}, k = 2, \ldots, m, \quad D := D_m,$$

and the departure process from scheduler k ($1 \leq k \leq m$) is related to the arrival process into that scheduler by

$$A_k(t) \geq D_k(t) \geq \inf_{s \leq t}[S_k(t-s) + A_k(s)], \qquad t \geq 0.$$

Define

$$S(t) := \inf_{\substack{t_1, \ldots, t_m \geq 0 \\ t_1 + \cdots + t_m = t}} \{S_1(t_1) + \cdots + S_m(t_m)\}, \quad t \geq 0.$$

The next theorem shows that the tandem of service curve schedulers can be replaced by a single scheduler with the service curve S.

Proposition 4.4 (Tandem of Schedulers as a Single Scheduler).

$$A(t) \geq D(t) \geq \inf_{s \leq t}[S(t-s) + A(s)], \qquad t \geq 0.$$

Combining this with the performance bounds for a single scheduler we readily obtain end-to-end performance bounds for a connection traversing a tandem of service curve schedulers. Note that the above composition rule is commutative and associative and hence the order of the schedulers in the tandem is irrelevant.

4.2 Closed Tandem of Schedulers

Now consider an closed tandem of m service curve schedulers as shown in Figure 3. This system differs from the open tandem in that the arrivals to the system do not get access to the first queue until there is a corresponding token present in the token bucket at the input to the tandem. Tokens are consumed by arriving packets that gain access to the first queue and are replenished by packets departing from the last queue. There are W tokens present in the token buffer initially. This system represents a connection that is window flow controlled with a window size of W. Note that the open tandem may be thought of as a special case with the window size $W = \infty$.

Let the arrival process to the closed tandem be denoted by A, the departure process by D, the arrival process into scheduler k by A_k and the departure process by D_k, $1 \leq k \leq m$. Then, the equations for a open tandem are also satisfied by the closed tandem with the important difference that now $A_1 = A \wedge (D_m + W)$ instead of $A_1 = A$. Let S be obtained by the same composition rule as for an open tandem. Based on the results for an open tandem we can reduce the closed cycle of m schedulers into a cycle of one scheduler with the service curve S.

Proposition 4.5.

$$A(t) \geq D(t) \geq \inf_{s \leq t}[S(t-s) + (A(s) \wedge (D(s) + W))], \qquad t \geq 0, \tag{4.6}$$

The main problem with the above characterization of the departure process from the closed cycle is that it involves an implicit inequality. We need to provide a way to construct a explicit lower bound on D. To this end we first consider the second inequality in (4.6) with an equality instead, i.e.,

$$\check{D}(t) = \inf_{s \leq t}[S(t-s) + (A(s) \wedge (\check{D}(s) + W))], \qquad t \geq 0. \tag{4.7}$$

Clearly, any solution to (4.7) also satisfies (4.6).

Theorem 4.6. *There exists a unique solution $\check{D}$ to* (4.7) *that is non-decreasing, and for any D satisfying* (4.6), $D \geq \check{D}$.

5. Regulators

In this section we study the use of regulators to shape or smooth the traffic of individual connections. We consider regulators that shape the traffic to any arbitrary upper envelope or burstiness constraint. Such regulators may be used at the point the traffic enters the network or before the schedulers at the output queues of the various switches along the connection's path. We show that the use of regulators which reshape the traffic to the original burstiness constraint of the connections arrival process leaves the end-to-end delay and backlog unchanged, while reducing the burstiness and therefore the buffer requirement at the intermediate schedulers (the buffering requirement is transferred to the regulators). Like our work on schedulers, the results in this section generalize existing results as well as provide more elegant proofs. The setup considered may once again be obtained from the SCFS model when combined with upper envelopes on the virtual departure process. Proceeding directly, consider a problem very similar to that of the synchronous map for the $\cdot/\cdot/1/\infty$ queue. Given an arrival process $A : \mathbb{R} \to \mathbb{R}_+$ with $A(t) = 0$ for $t < 0$ and a burstiness constraint or upper envelope $B : \mathbb{R}_+ \to \mathbb{R}_+$ (we assume that B is sub-additive and $B(0) = 0$) on the departure process, we seek to find the actual departure process $D : \mathbb{R} \to \mathbb{R}_+$ with $D(t) = 0$ for $t < 0$ satisfying the following conditions:

R1. $D(t) - D(s) \leq B(t-s)$ for all $s \leq t$.

R2. $D(t) \leq A(t)$ for all $t \geq 0$.

R3. D is the (pointwise) maximal function satisfying the above conditions.

As for the synchronous map, we can prove the following result giving an explicit solution for the above.

THEOREM 5.1. *There exits a unique D satisfying the above and is given by*

$$D(t) = \inf_{s \le t}[B(t-s) + A(s)], \qquad t \ge 0. \tag{5.1}$$

Note that this precise characterization of the departure process from a regulator satisfies the partial characterization of the departure process from a scheduler with a service curve constraint ((4.3) and (4.4)). Hence all results for a single scheduler or tandems of schedulers can be extended to a single regulator or a tandem or schedulers and regulators in arbitrary order. These include upper bounds on queue sizes, on the burstiness of the departure process, and on the delay. In addition, note that the second inequality in Proposition 4.4 may be seen to hold with equality in case we consider an open tandem or regulators alone and therefore such a tandem behaves behaves like a single regulator with a burstiness constraint given by the same composition rule as that for a tandem of schedulers. It is easy to verify that if the individual burstiness constraints are concave (and hence subadditive), then their pointwise minimum (which then is also concave and hence subadditive) is the burstiness constraint of the tandem. In general, the pointwise minimum is only an upper bound. This composition result can be used to obtain a implementation of a regulator with a complicated burstiness constraint as a tandem of regulators with simpler burstiness constraints.

References

[1] ANICK, D., MITRA, D., AND SONDHI, M. M. Stochastic theory of a data-handling system with multiple sources. *Bell Syst. Tech. J. 61* (1982), 1871–1894.

[2] BERAN, J., SHERMAN, R., TAQQU, M. S., AND WILLINGER, W. Long-range dependence in variable-bit-rate video traffic. *IEEE Trans. Comm. 43*, 2/3/4 (1995), 1566–1579.

[3] CRUZ, R. L. A calculus for network delay, part I: Network elements in isolation. *IEEE Trans. Inform. Theory 37* (1991), 114–131.

[4] CRUZ, R. L. Quality of service guarantees in virtual circuit switched networks. *IEEE Journal on Selected Areas in Communications 13* (Aug. 1995), 1048–1056.

[5] ELWALID, A. I., MITRA, D., AND STERN, T. E. Statistical multiplexing of Markov modulated sources: Theory and computational algorithms. In *Teletraffic and Data-traffic in a Period of Change, ITC-13*, A. Jensen and V. B. Iversen, Eds. Elsevier Science Publishers B. V. (North-Holland), 1991, pp. 495–500.

[6] HARRISON, J. M., AND REIMAN, M. I. Reflected brownian motion on an orthant. *The Annals of Probability 9*, 2 (1979), 302–308.

[7] PAREKH, A. K., AND GALLAGER, R. G. A generalized processor sharing approach to flow control in integrated services networks: The single node case. *IEEE/ACM Transactions on Networking 1*, 3 (June 1993), 344–357.

[8] PAREKH, A. K., AND GALLAGER, R. G. A generalized processor sharing approach to flow control in integrated services networks: The multiple node case. *IEEE/ACM Transactions on Networking 2*, 2 (Apr. 1994), 137–150.

[9] STILIADIS, D., AND VARMA, A. Latency-rate servers: A general model for analysis of traffic scheduling algorithms. In *Proc. IEEE INFOCOM'96* (San Francisco, Mar. 1996), pp. 111–119.

A Dynamic Scheduling Communication Protocol and its Analysis for Hypercubes *

Ayan Banerjee and Emmanouel (Manos) Varvarigos
University of California, Santa Barbara
Department of Electrical and Computer Engineering
Santa Barbara, CA 93106.

Abstract

We propose a new protocol for one-to-one communication in multiprocessor networks, which we call the Dynamic Scheduling Communication (or DSC) protocol. In the DSC protocol, the capacity of a link is partitioned into two channels: a data channel, used to transmit packets, and a control channel used to make reservations. We analyze the steady-state throughput of the hypercube implementation for random node-to-node communication, for various sizes of the hypercube. For the hypercube topology, and under the same set of assumptions on the node architecture and the routing algorithm used, the DSC protocol is found to achieve higher throughput than packet switching, provided that the size of the network is sufficiently large. We also investigate the relationship between the achievable throughput and the fraction of network capacity dedicated to the control channel, and present a method to select certain design parameters to optimize throughput.

1 Introduction

The *Dynamic Scheduling Communication* (DSC) protocol, proposed in this paper, is a reservation-based protocol for organizing the transmission of data in a multiprocessor network. In a typical VLSI implementation of a parallel computer, there are usually many wires connecting a pair of neighboring nodes. In the DSC protocol, a subset of these wires is allocated to the transmission of data packets, and the remaining wires are allocated to the transmission of control information, used to schedule future data.

Reservation-based communication protocols, such as the DSC protocol, are advantageous as they can provide lossless communication under heavy load conditions. Such protocols also avoid the waste of resources that arises when a packet is transmitted for several hops before being dropped (as may happen with packet switching, or the unavailability of resources that occurs when a packet is blocked at some intermediate link, preventing other packets from using that link (as may happen with wormhole routing [Dal90], unless virtual channels are used).

The most common example of a reservation-based scheme is circuit switching (see [GeR89], [ShV95]). Circuit switching, however, is inefficient for multiprocessors networks, because a link is reserved for more time than is required, and additional overhead is needed to "tear-down" a circuit when the transmission is completed. In another communication

*Research supported by NSF under grant NSF-RIA-08930554.

protocol based on reservations, the Conflict Sense Routing (CSR) protocol [VaB96], the time axis is divided into alternating control and data intervals, with reservations for data intervals being made during control intervals. A disadvantage of the CSR protocol is that the reservation overhead is completely determined by the network diameter and the length of the control and data packets, and neither is it flexible, nor is it under the control of the designer. The CSR protocol also requires the time-division multiplexing of data and control information on the same wires, which complicates implementation, and introduces additional overhead to distinguish between the two kinds of information. Deflection routing avoids the dropping of packets without using a reservation mechanism (see [GrG86], [GrH92], Kri[90]), but has several drawbacks [Max90], the most serious of which is that packets may arrive at their destination out of order, or they may circulate indefinitely without reaching their destination (live-lock) if special precautions are not taken. The Dynamic Scheduling Communication (DSC) protocol proposed in this paper, offers an alternative way to provide lossless communication with minimal buffer requirements, and it compares favorably to the previously mentioned protocols in terms of both performance and implementation complexity, provided that the overhead associated with reservations is kept low. Our results for the hypercube topology indicate that the throughput of the DSC protocol is superior to that of packet switching and the CSR protocol when the size of the network is sufficiently large. The DSC protocol is actually a family of protocols parameterized by an integer k, the optimal choice of which depends on the flit lengths, the packet lengths, and the diameter of the network. We provide a graphical method to find the optimal fraction of capacity that should be dedicated to the transmission of control information in order to maximize the throughput.

2 Description of the DSC protocol

In this section we present the DSC protocol for a general network topology, and we describe the data structures necessary for its implementation. The wires of a link that are dedicated to the transmission of data (or control) information form the *data* (or *control*, respectively) *channel* of the link. We let γ be the fraction of wires of a link that are used for the transmission of control information. The time axis in the control channel is divided into *control frames* of duration β (see Fig. 1), where β is a parameter that will be specified shortly. We assume that the network is synchronized so that frames start synchronously at all nodes, and that all packets have equal length and require one data slot in order to be transmitted over the data channel of a link. The duration of a data slot is equal to $\frac{L}{(1-\gamma)C} + t_m$, where L is the length of the packet in bits, t_m is an upper bound on the propagation and processing delay of a packet, and $(1-\gamma)C$ is the capacity dedicated to the transmission of data, measured in bits/sec. The duration β of a control frame is chosen to be equal to an integer number k of data slots. We also assume that a routing algorithm exists that selects a path for each source-destination pair, and provides a way to resolve conflicts among packets that request the same link.

A packet enters the network only after having reserved all the resources (links and buffer space) along its route. These reservations are made through the transmission of special control packets, called *flow control units* (or *flits*[1]), over the control channel. A flit requires one minislot in order to be transmitted over a link, with the duration of a minislot being equal to $\frac{F}{\gamma C} + t_f$, where F is the length of the flit in bits, and t_f is an upper bound on the propagation and processing delay of a flit on a link.

[1]Note that our definition of a flit differs from that given in [Dal90].

A control frame is divided into a *reservation phase* and a *confirmation phase*. During the reservation phase, a flit is sent towards the destination to reserve the required links and buffer space for those data slots during which the resource will be used[2]. A flit that

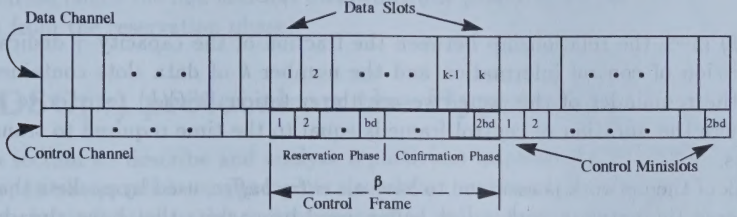


Figure 1: Control information is transmitted over a link by using a dedicated fraction γ of the capacity of the link. Each control frame is subdivided into control minislots as explained in the text. During a frame, k data packets can be transmitted along the data wires using the remaining $(1-\gamma)C$ of the capacity.

has been successful in making the appropriate reservations is sent (carrying an ACK), during the confirmation phase, on the reverse path from the destination to the source, in order to notify the source and confirm the reservations made at the intermediate nodes. A flit that was blocked at an intermediate node, because it could not make all necessary reservations, is also transmitted during the confirmation phase (carrying a NACK) from the intermediate node to the source, in order to notify the source of this failure, and release any resources that may have been reserved.

Flits can start transmission from a source only during the first control minislot of a control frame, and, as we will see, acknowledgements are guaranteed to return to the source by the end of that frame. A flit generated at a source follows the path determined by the underlying routing algorithm of the network. Reservations for links and buffer spaces are made for data slots that start after the termination of the control frame in which the flit is transmitted. To make sure that a (positive or negative) acknowledgement returns to the source by the end of the frame in which the flit is transmitted, we request that the duration β of a frame is (greater than or) equal to $2Bd$ minislots, where d is the diameter of the network, and B is the buffer size (measured in packets) per link. The durations of the reservation and the confirmation phase are both equal to $\beta/2$. To see that the reservation phase is completed in time $\beta/2$, note that if a flit is not blocked, it can be delayed by at most $B-1$ other flits on a given link during the reservation phase. This is because the buffer size per link is equal to B, and, if more than B flits request a link, at most B of them are allowed to proceed, and the remaining are blocked. Therefore, after time $\beta/2$, all flits have either arrived at their destinations or they have been blocked. To ensure that the confirmation phase is also completed in time $\beta/2$, we require that if a flit is transmitted on a link during minislot $i, i = 1, 2, \ldots, Bd$, of the reservation phase, it is transmitted on the same link (but in the reverse direction) during minislot $2Bd - i + 1$ of the subsequent confirmation phase. This ensures that flits do not collide during the confirmation phase, and feedback information arrives at the source reliably. A packet for which a NACK is received has to reattempt to make a reservation in one of the following frames.

The number of data slots k contained in a control frame is an important design pa-

[2]Resources are available for other packets for the remaining of the time.

rameter and is related to β as

$$\beta = 2Bd\left(\frac{F}{\gamma C} + t_f\right) = k\left(\frac{L}{(1-\gamma)C} + t_m\right). \quad (1)$$

Equation (1) gives the relationship between the fraction of the capacity γ dedicated to the transmission of control information and the number k of data slots contained in a frame. In the remainder of the paper we use the notation DSC(k) to refer to a DSC protocol where the duration of control frame is equal to the time required to transmit k data packets.

Each link of the network is assumed to have an *entry buffer*, used by packets that have not yet entered the network, and a *link buffer*, used by packets that have already been accepted in the network. A packet in an entry buffer sends out a flit, and, if it receives an ACK at the end of the frame, it enters the network at the beginning of the next frame by moving from the entry buffer to the corresponding link buffer. For every link queue $\mathcal{Q}_l$, there is a list $\mathcal{L}_l$, called *reservation list*. The t^{th} element of the list represents the t^{th} data slot relative to the start of the current frame. At the end of k data slots (or, equivalently, one control frame), the first k elements of $\mathcal{L}_l$ are deleted. The element $\mathcal{L}_l(t)$ that corresponds to data slot t consists of two fields, denoted by $\mathcal{L}_l(t)\_link$ and $\mathcal{L}_l(t)\_buffer$. The field $\mathcal{L}_l(t)\_link$ is equal to one, if link l has already been reserved for data slot t, and is equal to zero, otherwise. The field $\mathcal{L}_l(t)\_buffer$ takes integer values between zero and the buffer size B, and it is equal to the number of buffer spaces of $\mathcal{Q}_l$ that have already been reserved for data slot t.

Each flit f carries with it a counter c_f, which signifies the data slot (relative to the start of the current control frame) for which the next link is requested. The counter c_f is initially set to $k+1$. This is because a packet enters the network only after receiving a positive confirmation, which happens at the end of the frame during which the flit is sent. Therefore, the first link on the path should be requested for the first data slot that starts after the current frame is completed, which is data slot $k+1$. Flits arriving at a node are processed in the order they arrive with flits arriving during the same minislot processed arbitrary order.

We let c_f be value of the counter of flit f upon its arrival at link l, and $\mathcal{L}_l$ be the reservation list at the time flit f is processed at l. We also define T as the minimum integer satisfying (a) $c_f \leq T$, (b) $\mathcal{L}_l(T)\_link = 0$, and (c) $\mathcal{L}_l(t)\_buffer \leq B$, for all $t \in \{c_f, c_f+1, \ldots, T-1\}$. In other words, T is the first data slot, following the data slot for which the link is requested, for which no reservation has been made yet, and enough buffer space is available at the link to store the packet between its scheduled arrival and its departure from the link. If a T that satisfies the relations above does not exist, the flit f is blocked (the reservation fails). If, however, such a T exists, a reservation is made on l for data slot T, and the flit counter and the reservation list are updated according to (a) $c_f = T+1$, (b) $\mathcal{L}_l(T)\_link = 1$ and (c) $\mathcal{L}_l(t)\_buffer = \mathcal{L}_l(t)\_buffer + 1$, for all $t \in \{c_f, c_f+1, \ldots, T-1\}$. Note that buffer space is reserved for the packet at link l starting at data slot c_f (this is when the packet is scheduled to arrive at l) and ending at data slot T (this is the slot for which reservation on l is made for the packet). During the confirmation phase, all reservations made by flits that were eventually blocked are cancelled, and the corresponding resources are released. This can happen either explicitly (flits carrying NACKs follow the reverse path to the source freeing the links and buffer space), or implicitly (a resource is released if the reservation is not confirmed by the end of the frame). A packet for which an ACK is received at the end of a frame, enters the network at the beginning of the next frame, and is forwarded to its destination

using the links and buffer space that were reserved for it. The only additional control overhead required by the DSC protocol over packet switching is the counter c_f; the routing information carried by a flit does not have to be included in the packet itself when it is transmitted (since the link and slot over which the packet will be transmitted are already known from the reservation phase).

3 DSC(k) protocol on the Hypercube

In this section we describe and analyze a particular implementation of the DSC protocol in a hypercube parallel computer. We also obtain results on the fraction of capacity that needs to be dedicated to the control channel to optimize the throughput.

3.1 Implementation of the DSC(k) protocol in a hypercube

Each link of dimension i of a node s has an entry buffer $\mathcal{E}_i(s)$, which can store up to one packet originating at that link, and a queue $\mathcal{Q}_i(s)$, which is used to store transit packets. The entry buffer can accept a new packet only if the previous packet has reserved its entire path to the destination, and a positive feedback has been received. An entry buffer holding a packet for which no feedback has been received is said to be *backlogged*. New packets arriving at backlogged entry buffers are discarded.

The queue $\mathcal{Q}_i(s)$ is composed of two buffers, denoted by $\mathcal{Q}_i^0(s)$ and $\mathcal{Q}_i^1(s)$, each of which can hold one packet. Buffer $\mathcal{Q}_i^1(s)$ is used by packets that have the i^{th} bit of their routing tag equal to one. Buffer $\mathcal{Q}_i^0(s)$ is used by packets that do not have to cross the i^{th} dimension. Buffer $\mathcal{Q}_i^0(s)$ is connected to queue $\mathcal{Q}_{(i-1) \bmod d}(s)$ of the same node and buffer $\mathcal{Q}_i^1(s)$ is connected to queue $\mathcal{Q}_{(i-1) \bmod d}(s)$ of the neighbour node $s \oplus e_i$. This router architecture is called a *descending-dimensions switch*, and it leads to switches that are simpler, faster, and less expensive than crossbar switches (see [Dal90] and [VaB94]).

In our analysis we assume that there is buffer space only for the packet being transmitted (i.e, both $\mathcal{Q}_i^0(s)$ and $\mathcal{Q}_i^1(s)$ have buffer space $B = 1$); flits that find a link already reserved are blocked. When two flits attempt to make a reservation for the same (available) data slot during the same control frame, one of them is selected at random and the other is blocked. The number of minislots in the reservation phase or the confirmation phase is equal to the diameter d.

Flits that are successful in reserving all the links on the path to their destination, return during the confirmation phase to their origin, following the reverse path than the one followed in the reservation phase, and carry an ACK. A flit is transmitted over a link during control minislot $2d - i - 1, i = 0, 1, \ldots, d - 1$, of the confirmation phase if it was transmitted over the same link during control minislot i of the reservation phase; this guarantees that there are no collisions among flits during the confirmation phase. Flits that fail to make all the required reservations are discarded, and all unconfirmed reservations are cancelled at the end of a confirmation phase. Note that for the hypercube network and this particular implementation of the DSC protocol, each node needs to store at most two flits per link at any given time.

3.2 Throughput of the DSC(k) protocol for hypercubes

In this section we present an analysis of the DSC(k) protocol for a $d-$dimensional hypercube having descending-dimension switches. We assume that $d = kr$, for some integer r.

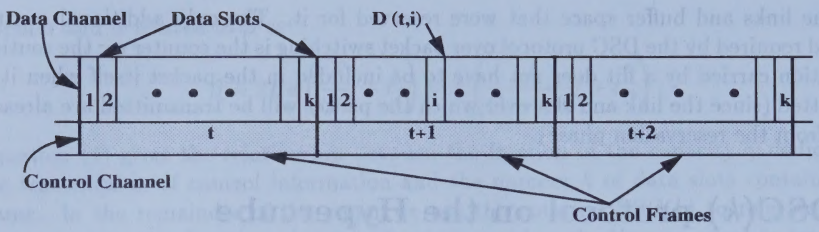


Figure 2: The i^{th} data slot following the end of frame t is denoted by $D(t,i)$. Note that with this notation, $D(t,k+i) = D(t+1,i)$.

It is possible for a flit to reserve a link and release it later, during the same control frame, due to its failure to reserve the remainder of the path. We refer to such a reservation as a ghost reservation, as opposed to a confirmed reservation, where a flit is successful in reserving all the links on its path. We define $p(t,i,l)$ as the probability that during control frame t, a reservation (ghost or confirmed) is made for link l for the i^{th} data slot following the end of frame t, where $i = 1,2,\ldots,d$. Assuming that the system eventually reaches steady state, the limit $p_i = \lim_{t\to\infty} p(t,i,l)$ exists, for all $i = 1,2,\ldots,d$, and is independent of the link l. Thus, p_i is the steady-state probability, that, in a given control frame, a link is reserved for the i^{th} data slot following the end of that frame. We also denote by p_0 that the entry buffer of a link is nonempty immediately prior to the beginning of a new control frame (i.e., a new or a retrial packet is available to enter the network at that particular link); we call p_0 the *attempt probability*. Since a flit is successful only after having reserved all the d links on its path, the average number of successful reservations made per link per frame is equal to p_d. Therefore the steady-state throughput R per node per data slot is $R = 2dp_d/k$. We evaluate p_d and then R by obtaining a recursive relationship on the steady-state probabilities p_i, $i = 1,2,\ldots,d$.

We denote by $D(t,i)$ the i^{th} data slot that follows the end of frame t (see Fig. 2). Note that with this notation, $D(t,k+i) = D(t+1,i)$. For a given link l, we let l_1 and l_2 be the internal and forward links, respectively, that lead to l. Link l may be reserved either by a flit f_1 arriving over link l_1, or by a flit f_2 arriving over link l_2. In particular, link l can be reserved during frame t for data slot $D(t,i)$ by a flit f_1 arriving over l_1 if (a) a reservation has been made by f_1 for link l_1 for the previous data slot $D(t,i-1)$; we call this event B, (b) link l lies on the path of f_1 (given that l_1 lies on its path); this happens with probability 1/2, and (c) no confirmed reservation has been made for link l for data slot $D(t,i)$ during previous control frames, and no reservation (confirmed or unconfirmed) has been made for $D(t,i)$ during the current control frame t by any flit arriving over link l_2. Thus, for $1 < i \leq d$, we have

$$p(t,i,l) = \frac{2 \times p(t,i-1,l_1)}{2}(1 - P(A/B)), \tag{2}$$

where A is the event that a confirmed reservation has been made on link l for data slot $D(t,i)$ during one of the previous frames $t-j, j = 1,2,\ldots,r-1$, or a reservation has been made for $D(t,i)$ during the current frame t by a flit arriving on link l_2. The probability that a flit f_2 requests link l during frame t for data slot $D(t,i)$ and is granted the link can be approximated by $\frac{1}{4} \cdot p(t,i-1,l_2)$. This is because for flit f_2 to reserve link l for data slot $D(t,i)$, it should have already reserved link l_2 for data slot $D(t,i-1)$. The factor of 1/4 is the probability that link l lies on the path of flit f_2 (given that l_2 lies on its path), and f_2 wins over flit f_1 when the conflict arises.

Since there is an attempt by flit f_1 arriving over l_1 to reserve link l for data slot $D(t,i)$, any confirmed reservations that have been made on link l for data slot $D(t,i)$ should have been made by flits arriving over link l_2. Hence, the probability that a confirmed reservation for $D(t,i)$ was made during a previous frame by a flit f_2 arriving over link l_2 is given by

$$\frac{1}{2}\sum_{j=1}^{n} p(t-j, i+jk-1, l_2) \cdot p(\text{flit } f_2 \text{ reserves } l \text{ given that it requests it }) \frac{p(t-j,d)}{p(t-j, i+kj, l)}, \tag{3}$$

where the number n of control frames during which reservation can be made for $D(t,i)$ is given by

$$n = \begin{cases} r-1-(i/k) & \text{if } i \bmod k \neq 0 \\ r-(i/k) & \text{if } i \bmod k = 0 \end{cases}.$$

The factor of $\frac{1}{2}$ is the probability that f_2 chooses link l, and the ratio $\frac{p(t-j,d)}{p(t-j,i+kj,l)}$ is the probability that the reservation made by f_2 was finally confirmed. To evaluate the probability that f_2 reserved link l during frame $t-j$ given that it requested it, note that f_2 would only lose out to a flit arriving over link l_1 that made a *ghost* reservation during frame $t-j$. This is because we know that a reservation is requested by flit f_1 for data slot $D(t,i)$, and, therefore, any flit that had reserved data slot $D(t,i)$ during frame $t-j$ coming from link l_1, should have eventually been blocked. Thus,

$$p(\text{flit } f_2 \text{ reserves } l \text{ given that it requests it }) = 1 - \frac{1}{4} p(t-j, i+jk-1, l_1)\left(1 - \frac{p(t-j,d)}{p(t-j,i+kj,l)}\right), \tag{4}$$

where $\frac{1}{4}$ accounts for the probability that a flit arriving over link l_1 choses link l and wins over a flit arriving over l_2. Taking the limit $t \to \infty$ and using the symmetry with respect to the links, Eqs. (2)-(4) give

$$p_i = p_{i-1}\left(1 - \frac{p_{i-1}}{4} - \frac{1}{2}\sum_{j=1}^{n}\left[p_{i+jk-1}\frac{p_d}{p_{i+kj}}\left(1 - \frac{p_{i+jk-1}}{4}\left(1 - \frac{p_d}{p_{i+jk}}\right)\right)\right]\right) \quad \text{for } i = 2, \ldots, d. \tag{5}$$

In order to relate p_1 with p_0, we observe that a link l may be reserved for the first data slot of a given frame during one of the $r-1$ frames that precede it. During each of these frames, a confirmed reservation on l for that data slot is made with probability p_d. Since confirmed reservations made for the same link and data slots are disjoint events, the probability that the link is unreserved is $1-(r-1)p_d$, and we have

$$p_1 = p_0(1-(r-1)p_d). \tag{6}$$

Equation (5) can be solved to give p_{i-1} in terms of $p_j, j \geq i$. A curve can then be plotted for the average throughput $R = 2dp_d/k$ per node per data slot as a function of the attempt rate p_0, by choosing a particular value of p_d and then successively obtaining $p_{d-1}, p_{d-2}, \ldots, p_0$, by running a backward recursion.

3.3 Analytical and Simulation Results

Figures 3(a) and 3(b) compares the analytical and simulation results obtained for the average throughput per node and data slot of the DSC(k) protocol on hypercubes of dimension 6 and 8 for several values of k, assuming that there is no buffer space at the

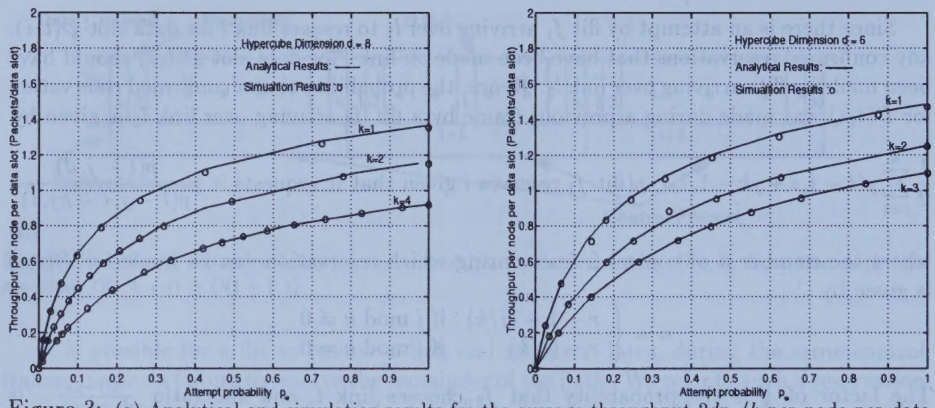


Figure 3: (a) Analytical and simulation results for the average throughput $2dp_d/k$ per node per data slot of the hypercube implementation of the DSC(1), DSC(2) and DSC(4) protocols for dimension $d = 8$. (b) Analytical and simulation results for the average throughput of the hypercube implementation of the DSC(1), DSC(2) and DSC(3) protocols for dimension $d = 6$. Note that the duration of a data slot is not the same for the different versions of the DSC(k) protocol, since it depends on the value of k (see also Fig. 6).

nodes, except for the packets currently under transmission. The difference between analytical and simulation results is consistently smaller than 2%. Note that the maximum throughput that can be achieved in a hypercube for uniform distribution of the destinations assuming infinite buffer space per node is at most equal to 2 packets per node and data slot.

Since (by Eq. (1)) the duration of a data slot is different for different values of k, the results given in Fig. 3 are not directly comparable. This is because they do not take into account the control overhead, which is larger for small values of k. To compare the performance of the DSC(k) protocol for different values of k, we define the *normalized throughput* as the percentage of the total capacity that is efficiently used (capacity is not used efficiently if it is used to transmit control bits, or if it stays idle). In Fig. 4(a) we plot the normalized throughput of the DSC protocol for $d = 8$ and several values of k, as a function of the ratio $\frac{F}{L}$ of the length F of a flit over the length L of a data packet, assuming $t_m \approx t_f \approx 0$. In Fig. 4b we also plot the fraction γ of the capacity allocated to the transmission of control information as a function of $\frac{F}{L}$ for several values of k. We note that with increasing ratio $\frac{F}{L}$, it is more advantageous to use larger values of k.

Given the values of the parameter $\frac{F}{L}$ and the hypercube dimension d, it is possible to find the value of k (or equivalently, the fraction γ of capacity to be allocated to the control channel) that maximizes the normalized throughput. For example, for hypercube dimension $d = 8$, flit size $F = 64$, and packet size $L = 2048$ (in bits), Fig. 4a shows that the normalized throughput is maximized when $k = 2$ (for these values of F and L, we have $\frac{F}{L} = 0.03125$). From Fig. 4b we see that this corresponds to $\gamma = 0.2$, which implies that in the VLSI design we should allocate 1 in every 5 wires of a link to the transmission of control information. In the DSC protocol, the control overhead is thus determined by the ratio of the control wires to the total number of wires, and it can be chosen so as to maximize the throughput. This is not possible with the CSR protocol [VaB96], where the designer does not have such flexibility. In general, for different values of the parameters F, L and d, a different version of the DSC(k) protocol maximizes the normalized throughput. For example, if $L = 2048$ and $d = 8$, the DSC(2) protocol is

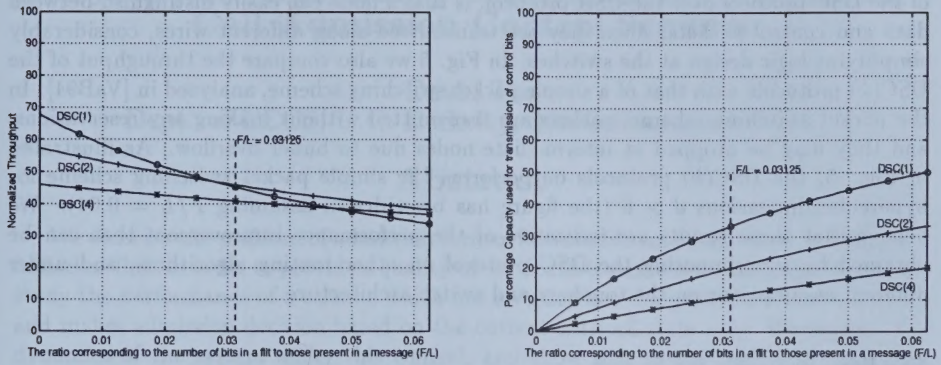

Normalized Throughput
DSC(1)
DSC(2)
DSC(4)
F/L = 0.03125
The ratio corresponding to the number of bits in a flit to those present in a message (F/L)
Percentage Capacity used for transmission of control bits
F/L = 0.03125
DSC(1)
DSC(2)
DSC(4)
The ratio corresponding to the number of bits in a flit to those present in a message (F/L)

Figure 4: (a) The normalized throughput as a function of $\frac{F}{L}$ for a 8−dimensional hypercube and several values of k (or equivalently γ). (b) We also illustrate the variation of the fraction γ of the capacity dedicated to the control channel, obtained by Eq. (2) as a function of $\frac{F}{L}$ and k.

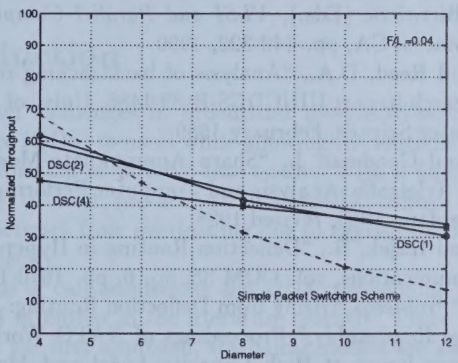


Figure 5: Illustrates (a) the normalized throughput of the DSC(k) protocol for different values of the parameter k as a function of the dimension d of the hypercube, for $F = 64$ and $L = 1600$, (b) the throughput obtained by the simple packet switching scheme .

optimal for $F > 57$ bits, while the DSC(1) protocol is optimal for $F < 57$ bits.

3.4 Comparison with Packet Switching and the CSR protocol

In Fig. 5, we illustrate the normalized throughput of the DSC(k) protocol for different values of k as a function of the dimension d of the hypercube. For a fixed k, the normalized throughput per node decreases when the dimension d increases. This is mainly due to the descending-dimension switches used, and has also been observed in other works (see [Dal90], [VaB94]). Note that for hypercubes of larger dimensions, larger values of k are preferable. For example, when $F = 64$ and $L = 1600$, protocol DSC(1) is better for $d = 4$, DSC(2) is better for $d = 8$ and 12, and DSC(4) is better for $d \geq 16$. In general we expect that the larger the diameter of a network is, the larger the parameter k should be in order to obtain the optimal balance between the capacity allocated to the control channel and the capacity available for the transmission of data.

Since the CSR protocol [VaB96] can be viewed as the time division multiplexed version of DSC(1), its performance is inferior to that of the (optimized) DSC protocol, when the diameter of the network is large, or when the ratio F/L is large. An additional advantage

of the DSC protocol over the CSR protocol, is that a node can easily distinguish between data and control packets, since they are transmitted along different wires, considerably simplifying logic design at the switches. In Fig. 5 we also compare the throughput of the DSC(k) protocols with that of a simple packet switching scheme, analyzed in [VaB94]. In the packet switching scheme, packets are transmitted without making any reservations, and they may be dropped at intermediate nodes due to buffer overflow. As illustrated in Fig. 5, the DSC(k) protocols outperforms the simple packet switching scheme for hypercube dimensions $d > 6$ (the figure has been drawn assuming $F/L = 0.04$). We believe that these results are indicative of the performance improvement that can be obtained by superimposing the DSC protocol on other routing algorithms, and under different assumptions on the topology and switch architecture.

References

[Dal90] Dally, W.J., " Network and Processor Architecture for Message-Driven Computers," in R. Suaya, and G. Birtwistle (Eds.), *VLSI and Parallel Computation*, Morgan Kaufmann Publishers, San Mateo, CA, pp. 140-222, 1990.

[GeR89] Grunwald, D.C., and Reed, D.A., "Analysis of backtracking routing in binary hypercube Computers," Research Report UIUCDCS-R-89-1486, Univ. of Illinois, Urbana-Champaign, Dept. of Computer Science, February 1989.

[GrG86] Greenberg, A.G., and Goodman J., "Sharp Approximate Models of Adaptive Routing in Mesh Networks," Teletraffic Analysis and Computer Performance Evaluation, pp. 255-270, Elsevier, Amsterdam, 1986, revised 1988.

[GrH92] Greenberg, A.G., and Hajek, B., "Deflection Routing in Hypercube Networks," IEEE Transactions on Communications, vol. COM-35, no. 6, pp. 1070-1081, June 1992.

[Max90] Maxemchuk, N.F., "Problems Arising from Deflection Routing: Live-lock, Lock-out, Congestion and Message Reassembly," Proceedings of NATO Workshop on Architecture and High Performance Issues of High Capacity Local and Metropolitan Area Networks, France, June 1990.

[Kri90] Krishna A., "Communication with Few Buffers: Analysis and Design," Ph.D. Thesis, Department of ECE, University of Illinois at Urbana-Champaign, December 1990.

[ShV95] Sharma, V., and Varvarigos, E.A., "Circuit switching with input queueing: An analysis for the $d-$dimensional wraparound mesh and the hypercube," to appear IEEE Transactions on Parallel and Distributed Systems.

[VaB94] Varvarigos, E.A., and Bertsekas, D.P., "Performance of Hypercube Routing Schemes With or Without Buffering," IEEE/ACM Transactions on Networking, vol. 2, no. 3, pp. 299-311, 1994.

[VaB96] Varvarigos, E.A., and Bertsekas, D.P., "A Conflict Sense routing Protocol and its Performance for Hypercubes," IEEE Transactions on Computers, vol. 45, no. 6, pp. 693-703, June 1996.

Asymptotic Optimality of a Measurement-Based Call Admission Control Scheme

David Tse
Dept. of EECS, U.C. Berkeley, dtse@eecs.berkeley.edu

Abstract

We consider the problem of admission control for variable-rate traffic sources sharing a bufferless link, in order to provide a quality-of-service in terms of overload probability. We study the performance of a scheme which has no prior knowledge of the traffic statistics and makes admission decision based on the current network state only. We analyze the dynamics of the system under this control, and show that in the regime of large link capacity and separation of call and burst time-scales, this scheme performs as well as the optimal scheme which has full knowledge of the statistics.

1 Introduction

One of the main challenges in broadband integrated-services networks is to provide quality-of-service (QOS) to users while allowing the efficient sharing of resources through statistical multiplexing of variable-rate traffic. With a fixed amount of resources and a target QOS level to meet, pro-active call admission control should be exercised in order to prevent the network from overloading due to aggregrate bandwidth demand exceeding the network capacity.

The problems of call admissions and QOS provisioning for variable-rate traffic have received a lot of attention in the literature recently. There are two main approaches to the problem. In the first approach, statistical models are assumed for the traffic sources and admission control done based on overload probability estimate using the statistical model. This approach focuses on computing the *effective bandwidth* of variable-rate traffic sources, which characterizes the amount of resources a call needs for a desired QOS (see e.g [Hui88]). The problem of this approach is that it is often difficult to obtain accurate statistical model *a priori* for many traffic sources, and it is unclear how sensitive the performance of the admission controller on the inaccuracy of such source model.

In the second approach, stochastic models for sources are forgone. Instead, traffic sources are assumed to be tightly policed by descriptors such as leaky buckets, which constrains the long term mean rate and the burst size , and admission control is done based on the worst case regulated behavior of the sources. (see e.g. [Cru91]) This approach suffers from the problem of being in general conservative. More importantly, it is generally difficult to choose a static descriptor which fits the behavior of the source well throughout the duration of the call. For traffic like compressed video with significant slow time-scale fluctuations, the leaky bucket parameters will either have to be chosen with very large bucket size or the token rate close to the peak rate of the source. Admission control based on these parameters will result in a severe under-utilization of network resources.

In this paper, we focus on a third approach, in which admission control decisions are made based on network measurements alone. Instead of assuming a statistical model for the traffic or a worst-case regulated behavior of the sources, the past behavior of the

current calls are monitored and the information is used to make decisions to accept or reject new calls. This measurement-based approach alleviates the burden on the users to provide accurate statistical model or to find good traffic descriptors, and thus is a more practical approach for achieving statistical multiplexing gain with variable-rate traffic.

A fundamental issue in the design of any measurement-based call admissions control scheme is the amount of history to use. If call statistics are stationary, then the performance of a measurement-based admission control scheme will always improve with the amount of history it uses in the estimation of the overload probability. In fact, as more statistics are gathered, the performance of the scheme will approach one which knows the statistics of the bandwidth requirements of the calls in advance. In reality, the system is not stationary over an arbitrarily long time-scale. For example, the nature of calls changes over time. On the other hand, it is also clear that some amount of stationarity is needed for any measurement-based scheme to work.

Our approach towards understanding this issue is to assume a stationary model and use it to evaluate the performance of admission control schemes which have limited memory. In this paper, we will focus on the extreme case, where admission decisions are made based on the current state of the network only. We call such schemes *memoryless*. We view this as the first step towards understanding the performance gain from increasing the amount of memory. The general goal is to find a call admission scheme with sufficient memory to match the performance target when the system is assumed stationary, but little enough memory to achieve good reactivity to non-stationarities.

The specific admission control scheme we will be investigating is of the *certainty equivalent* type. Namely, the admission controller assumes that the measured statistics are the true statistics of the calls, and uses the information to make decisions in the same way as a controller which has perfect *a priori* knowledge of the call statistics would. Certainty-equivalent controllers are generally sub-optimal, as they do not take the *measurement errors* into account.

Despite the simplicity of the scheme, we will show that in the regime of large link capacities and separation of call and burst time-scale, a memoryless certainty-equivalent control can achieve the performance of an optimal scheme tuned to the traffic statistics. The proof of this result yields the important insight that it is essential to consider the *dynamics* of the controlled system to gain a full understanding of its performance. Thus, even though the controller is prone to measurement errors at any single admission decision, it turns out that in the above parameter regime, overload occurs only after a *succession* of admission mistakes and this event is much more unlikely.

Past work on measurement-based admission control [CK], [SS91], [HW] have either ignored measurement errors or assumed a static situation where calls do not arrive or depart the system and there is arbitrarily long time to make accurate measurements. Recent work by Gibbens et. al. [GKK95] advocates the explicit incorporation of call-level dynamics into the model and provides much inspiration to the present work. However, while they focus on the use of prior knowledge of the statistics in a decision-theoretic framework, we will show that even a very simple scheme without any prior knowledge can achieve asymptotic optimality.

2 Analysis

2.1 Basic Model

The network resource is a single bufferless link with capacity C. Calls arrive according to a Poisson process at rate λ and stay for an exponential distributed time with mean T if admitted. The calls have identical bandwidth requirement statistics, described by a continuous-time ergodic Markov fluid process. There are K states in the Markov process, and a call generates fluid at rate of μ_k in state k. The transition rate from state k to state l is r_{kl}. Let π_k be the steady-state probability that the source is at state k. Also, assume that when a call begins to transmit data, the Markov process is at steady state. The fluid processes of different calls are assumed to be independent.

A measurement-based admission control scheme decides whether to accept a new call based on the observed past history of calls that are currently in the system and possibly those that have already departed the system. Note that given any admission control scheme, the entire system is a stationary process. We are interested in two performance measures of a scheme: the steady-state probability of the event that the system overloads, i.e. the instantaneous aggregate fluid rate of calls in the system exceeds C, and the expected fraction of the total bandwidth utilized. For a given arrival rate, maximizing the latter is equivalent to minimizing the blocking probability. The success of the admission control scheme is evaluated by how well it meets the QOS-requirement (in terms of overload probability) and how close its bandwidth utilization is to that of the optimal scheme which knows the bandwidth requirements statistics *a priori*.

Suppose the statistics of each call is known a priori. Then one can estimate the overload probability given the number of calls in the system. For large number of calls, an accurate approximation is the Chernoff's estimate: the probability of overload when there are m calls in the system is approximately

$$\exp(-mL^*(\frac{C}{m})) \tag{1}$$

where

$$L*(\mu) = \sup_{r>0}[\mu r - L(\mu)], \qquad L(r) = \log(\sum_{i=1}^{K} \pi_k \exp(\mu_k r))$$

assuming that the average rate $\bar{\mu}$ of each call is less than $\frac{C}{m}$ so that overloading is indeed a rare event. This is the well known approximation for the tail of the distribution of the sum of n i.i.d. random variables having the stationary bandwidth requirement distribution of a call. To satisfy a desired overload probability p_{qos}, one can *a priori* compute the maximum number of calls m^* such that the above approximation is less than p_{qos}. The admission control is simply to accept a new call if there are less than m^* calls in the system, and reject it otherwise. The notion of *effective bandwidth* for bufferless system is developed using this approximation [Hui88]; in this homogeneous case, the effective bandwidth of a call is simply $\frac{C}{m^*}$.

Now we should focus on the problem of interest, when no prior information about the statistics of the calls is available. The idea now is to estimate the statistics of the calls from observing their past empirical behavior. We will study a very simple scheme to focus on the essential issues. Motivated by the recent work of Gibbens et al. [GKK95] on measurement-based admission control, we consider a scheme which is *memoryless*, i.e. every time a new call arrives, the scheme uses only information about the *current* state

of the network in making the decision of accepting or rejecting the call. More specifically, the scheme determines the number of calls $n_k(t)$ that is currently generating data at rate c_k, for each k ($k = 1, \ldots, K$). This yields an empirical distribution $\{\hat{\pi}_k\}$ of bandwidth requirements for a typical call, where $\pi_k = \frac{n_k(t)}{x(t)}$ and $x(t)$ is the number of calls currently in the system at time t. The idea is to use $\{\hat{\pi}_k\}$ to estimate the distribution $\{\pi_k\}$ of the bandwidth requirements throughout the entire lifetime of a call. The admission criterion for the new call for a given threshold p_{qos} on the renegotiation failure probability is taken to be:

$$\exp(-\hat{L}^*(\frac{C}{x(t)+1}) \cdot (x(t)+1)) \leq p_{qos} \qquad \text{and} \qquad \sum_{k=1}^{K} \hat{\pi}_k \mu_k < C \tag{2}$$

where

$$\hat{L}(r) = \log \sum_{k=1}^{K} \hat{\pi}_k \exp(\mu_k r) \qquad \hat{L}^*(\mu) = \max_{r>0}[\mu r - \hat{L}(r)]$$

Note that the left hand side of inequality (2) is the Chernoff approximation of the failure probability for a system with $n+1$ calls each with bandwidth requirements distributed as $\{\hat{\pi}_k\}$. Thus, this admission control scheme is of the certainty-equivalent type: the controller assumes that the measured values are the true parameters and acts like the optimal controller which has perfect knowledge of the values of those parameters.

2.2 Asymptotic Analysis

To get some insights into the dynamics of this scheme and to compare its performance to the scheme with perfect knowledge, we will consider a fluid approximation as well as a large deviations analysis for the above model. Such an analysis is relevant in the asymptotic regime of large link capacity and small loss probabilities.

Let the total capacity be scaled as $C \equiv n$, and the call arrival rate as $\frac{\lambda n}{T}$, where n is a large parameter. Thus, $n\lambda\bar{\mu}$ is the offered load, and $\lambda\bar{\mu}$ is the offered utilization, i.e. offered load normalized by the the link capacity. Keeping the ratio of offered load to system capacity fixed in this scaling means that we are assuming that the link capacity is sized so that it is of the same order of magnitude as the demand. Also, note that the offered load is *independent* of the average call holding time. We also set the quality-of-service requirement $p_{qos} = \exp(-n\delta_{qos})$, for a parameter δ_{qos} which specifies the exponential decay rate. Thus, the larger the system, the more stringent is the QOS requirement,

To explain ideas in the simplest terms, let us first focus on two-state on-off sources. When it is on, the source transmits fluid at the peak rate of 1; when it is off, rate 0. Let p be the steady-state probability that a source is on, i.e. its mean rate is p. By an appropriate normalization of the time unit, we can assume that the on-to-off transition rate is $1-p$ and the off-to-on transition rate is p.

2.2.1 Known Statistics

Consider first the case when admission control is done using the Chernoff's estimate eqn. (1) of the overload probability based on perfect prior knowledge of the source statistics. For the n th system, let $G_n^*(t)$ and $H_n^*(t)$ be the total number of calls in the system and the number that are on at time t respectively. (The superscript '*' denotes quantities associated with scheme with perfect prior knowledge.) Define the scaled processes:

$$X_n^*(t) \equiv \frac{G_n^*(t)}{n}, \qquad Z_n^*(t) \equiv \frac{H_n^*(t)}{n}$$

and let $u^*(n,T) \equiv \mathbb{E}[Z_n^*(t)]$ be the average utilization.(We indicate explicitly the dependence on the mean call holding time T.). Also, let $p^*(n,T)$ be the overload probability.

For the nth system, the maximum number of calls $m^*(n)$ admitted under the admission control is the largest m such that

$$mL^*(\frac{n}{m};p) \geq n\delta_{qos} \qquad \text{and} \qquad \frac{n}{m} \geq p$$

where L^* is large deviations rate function for on-off sources with mean rate p. It can be explicitly calculated as:

$$L^*(\mu;p) = \mu \log(\frac{\mu}{p}) + (1-\mu)\log(\frac{1-\mu}{1-p}) \tag{3}$$

for $\mu \in [0,1]$ and equals ∞ otherwise. Note that $L^*(1;p) = -\log p$, correspond to the exponent of the probability that all sources are on.

As $n \to \infty$, it can be seen that $\frac{m^*(n)}{n}$ converges to $x^*(\delta_{qos})$ which is the largest $x \in [1, \frac{1}{p}]$ satisfying:

$$xL^*(\frac{1}{x};p) \geq \delta_{qos}$$

or equivalently,

$$-\log(xp) + (x-1)\log\frac{x-1}{x-xp} \geq \delta_{qos}$$

If $\delta_{qos} \leq -\log p$, then $x^*(\delta_{qos})$ is the unique solution to the equation

$$-\log(xp) + (x-1)\log\frac{x-1}{x-xp} = \delta_{qos}$$

On the other hand, if $\delta_{qos} > -\log p$, then $x^*(\delta_{qos}) = 1$. In this case, the QOS requirement is too stringent (relative to the burstiness of the traffic) such that peak-rate admission control has to be done. The system will never overload but also no statistical multiplexing gain is possible.

We now examine the behavior of the system as $n \to \infty$. It can be seen that typical behavior of the scaled system is well approximated by a *fluid limit* in which calls arrive at a deterministic rate of $\frac{\lambda}{T}$ and they transmit date at the mean rate of p once they are admitted into the system. The expected utilization of the system in this fluid limit depends on the offered rate λ:

$$\lim_{n\to\infty} u^*(n,T) \equiv u_\infty^* = \begin{cases} \lambda p & \text{if } \lambda < x^*(\delta_{qos}) \\ x^*(\delta_{qos})p & \text{if } \lambda > x^*(\delta_{qos}) \end{cases}$$

where $x^*(\delta_{qos})$ is computed as above. In the first case, the offered load is sufficiently low such that almost all calls are admitted. In the second case, the offered load is too high so that to maintain the desired QOS, a fraction of the offered calls has to be rejected. In this case, the number of calls in the system most of the time is close to the maximum possible for a given QOS. Moreover, since the number of calls in the system is no greater than

$x^*(\delta_{qos})$ at all times, it is clear that by definition of $x^*(\delta_{qos})$, that the QOS is satisfied, i.e. for all T,

$$\lim_{n\to\infty} \frac{1}{n} \log p^*(n,T) \leq -\delta_{qos}$$

. In particular, in the case when $x^*(\delta_{qos}) = 1$, $p^*(n,T) = 0$ for all n and T. In the case when $x^*(\delta_{qos}) > 1$ and $\lambda > x^*(\delta_{qos})$, the QOS is satisfied exactly (in an exponential sense) for large call holding time T, i.e.

$$\lim_{T\to\infty} \lim_{n\to\infty} \frac{1}{n} \log p^*(n,T) = -\delta_{qos}$$

2.2.2 Unknown Statistics

We now turn to an asymptotic analysis of the memoryless admission control scheme. In analogous to the previous analysis, for the nth system, let $G_n(t)$ and $H_n(t)$ be the total number of calls in the system and the number that are on at time t respectively, under the memoryless admission control scheme. Note that $(G_n(t), H_n(t))$ is the state of the system at time t. Define the scaled processes:

$$X_n(t) \equiv \frac{G_n(t)}{n}, \qquad Z_n(t) \equiv \frac{H_n(t)}{n}$$

and let $u(n,T) \equiv \mathbb{E}[Z_n(t)]$ be the average utilization. Also, let $p(n,T)$ be the overload probability. Our goal is to compare the average utilization and the overload probability to the corresponding quantities under the scheme with perfect prior knowledge, for n grows large.

The measurement-based scheme estimates the statistics of the sources under their current bandwidth requirements. In the case of on-off sources, the only parameter to estimate is p, the mean bandwidth requirement. The estimate $\hat{p}(t)$ of p at time t is $\hat{p}(t) = \frac{H_n(t)}{G_n(t)}$. Using the certainty-equivalent admission criterion (2), a call is admitted at time t if and only if

$$L^*(\frac{n}{G_n(t)+1}; \hat{p}(t)) \geq n\delta_{qos} \qquad \text{and} \qquad H_n(t) < n \tag{4}$$

where L^* is given by eqn. (3).

The scaled process $(X_n(t), Z_n(t))$ is a jump Markov process on the state space $\mathcal{S}^{(n)} \equiv \{(\frac{i}{n}, \frac{j}{n}) : 0 \leq j \leq i\}$. Define the sets

$$\begin{aligned} \mathcal{S}_a^{(n)} &\equiv \{(\frac{i}{n}, \frac{j}{n}) \in \mathcal{S}^{(n)} : L^*(\frac{n}{i+1}; \frac{j}{i}) \geq n\delta_{qos} \qquad \text{and} \qquad \frac{j}{n} < 1\} \\ \mathcal{S}_r^{(n)} &\equiv \mathcal{S}^{(n)} - \mathcal{S}_a^{(n)} \end{aligned}$$

New calls are admitted if the state of the system is in $\mathcal{S}_a^{(n)}$ and rejected if in $\mathcal{S}_r^{(n)}$. For large n, the scaled processes $(X_n(t), Z_n(t))$ converge to a fluid limit which take values in a two dimensional continuous state space $\mathcal{S} \equiv \{(x,z) : 0 \leq z \leq x\}$. The regions $\mathcal{S}_a^{(n)}$ and $\mathcal{S}_r^{(n)}$ converge into two regions in $\mathcal{S}$ divided by a boundary. Define f on $\mathcal{S}$ as

$$f(x,z) = \begin{cases} -\log z + (x-1)\log\frac{x-1}{x-z} & \text{if } x \geq 1 \\ \infty & \text{else} \end{cases}$$

Then in the region $\mathcal{S}_a \equiv \{(x,z) \in \mathcal{S} : f(x,z) \geq \delta_{qos}, z < 1\}$, calls are accepted where in the complement $\mathcal{S}_r$, calls are rejected. We shall call $\mathcal{S}_a$ the *acceptance region* and $\mathcal{S}_r$ the *rejection region.* It should be noted that this partitioning depends both on the number of sources x in the system and the number z which are on, whereas in the case with known source statistics, it depends only on the number of sources x which are on (i.e. whether $x \geq x^*(\delta_{qos})$). This is the essence of the difference between a measurement-based scheme and one with known statistics.

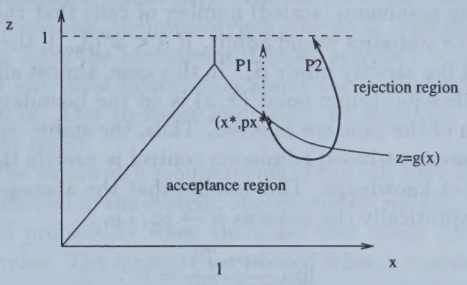


Figure 1: State space of fluid limit.

We shall now gather more information about the boundary. Clearly, if $(x,z) \in S_a$ if $x \leq 1$. (No possibility of overload even if all sources in the system transmit at peak rate.) For $x \geq 1$, $(x,z) \in S_a$ if $f(x,z) \geq \delta_{qos}$ and $z < 1$. Further analysis reveals that one can find a function g such that this condition is equivalent to $z \leq g(x)$, where g satisfies $f(x,g(x)) = \delta_{qos}$ and has the properties that:1) $\lim_{x \to 1+} g(x) = \exp(-\delta_{qos})$; 2) $g(x)$ decreases monotonically with x; and 3) $\lim_{x \to \infty} g(x) = z^*$ where $z^* > 0$ satisfies

$$z^* - \log z^* - 1 = \delta_{qos}$$

Note that when there are x sources, $\frac{g(x)}{x}$ is the mean rate of each source such that the overload probability requirement is just satisfied. For the number of sources x close to 1 (link capacity), the only way to have an overload event is for all sources to transmit at peak rate. To satisfy the desired QOS, the mean rate of each source should then be $\exp(-\delta_{qos})$. Properties (2) and (3) imply that as there are more sources in the system, the maximum number of them that can be on and still permits new admissions decrease, and moreover approaches a non-zero limit as x goes to infinity. The situation is depicted in Figure (1). Note that the boundary consists of two segments: $z = g(x)$ for $x \in (1, \infty)$ and $x = 1$ for $z \in (\exp(-\delta_{qos}), 1)$. Also, the boundary partitions the state space into two connected regions.

In the fluid limit, the process follows a deterministic vector field starting from any initial point. In region $\mathcal{S}_a$, calls are accepted into the system at deterministic rate $\frac{\lambda}{T}$, while in region $\mathcal{S}_b$, the acceptance rate is zero. We now find the stable equilibrium points of the vector field and thus compute the average utilization in the fluid limit. First, one can easily see that the stable equilibrium points must lie on the line $z = px$, since p is the (true) mean rate of each source. As a consequence of the law of large numbers, the typical fraction of sources in the system that are on must be p. Moreover, there are no stable equilibrium points strictly inside the rejection region $\mathcal{S}_r$, since in this region,

sources leave the system and no new sources enter. Thus, the stable equilibrium points must be strictly inside $\mathcal{S}_a$ or on the intersection of the boundary and the line $z = px$.

Suppose the line $z = px$ intersects the boundary at the point $(x^*(\delta_{qos}), px^*(\delta_{qos}))$. If $\delta_{qos} \leq -\log p$, the intersection is with the segment $z = g(x)$, and $x^*(\delta_{qos}) \geq 1$ satisfies the equation:

$$-\log(xp) + (x-1)\log\frac{x-1}{x-xp} = \delta_{qos}$$

If $\delta_{qos} > -\log p$, the intersection is with the segment $x = 1$, and $x^*(\delta_{qos}) = 1$. Note that $x^*(\delta_{qos})$ is precisely the maximum (scaled) number of calls that the scheme with perfect knowledge of the source statistics would admit. If $\lambda < x^*(\delta_{qos})$, the (unique) stable point $(\bar{x}, \bar{z})$ is $(\lambda, \lambda p)$, which lies strictly inside $\mathcal{S}_a$. In this case, almost all calls are accepted. If $\lambda \geq x^*(\delta_{qos})$, the stable equilibrium point $(\bar{x}, \bar{z})$ is on the boundary: $(x^*(\delta_{qos}), px^*(\delta_{qos}))$. In this case, a fraction of the calls are rejected. Thus, the stable equilibrium for the fluid limit under the measurement-based admission control is *exactly* the same as that under the scheme with perfect knowledge. This implies that the average utilization under the two schemes are asymptotically the same as $n \to \infty$, i.e.

$$\lim_{n\to\infty} \frac{u(n,T)}{u^*(n,T)} = 1 \tag{5}$$

We now turn to an analysis of the overload probability $p(n,T)$ under the memoryless admission control scheme. This corresponds to the event $\mathcal{O} = \{(x,z) : z > 1\}$. Since this set does not contain the stable equilibrium, it is a rare event for n large, with probability decaying exponentially with n. We have:

$$p(n,T) \approx \exp[-nI^*(T)] \tag{6}$$

where

$$I^*(T) = \inf_{(\vec{r}(\cdot),\tau)\in F} \int_0^\tau l(\vec{r}(t), \dot{\vec{r}}(t))dt \qquad \text{and} \quad F = \{(\vec{r}(\cdot),\tau) : \vec{r}(0) = (\bar{x},\bar{z}), \vec{r}(\tau) \in \mathcal{O}\} \tag{7}$$

Thus, the exponent is given by the cost of the cheapest path starting from the equilibrium point and leading to an overload. This is in accordance with *Laplace's* principle: the probability of the most likely way for a rare event to happen is approximately the same as the probability of the rare event. Here, $l(x,z,\vec{v})$ is the *local rate function*, which gives the cost of traveling with velocity $\vec{v}$ at the state (x,z).

At present, we do not have a rigorous large deviations limit theorem justifying the approximation (6); thus, it is only a heuristic. To have such a theorem, we need to prove a large deviations principle for the scaled processes $(X_n(\cdot), Z_n(\cdot))$, which have discontinuous jump rates across a *curved* boundary. Dupuis and Ellis [DEed] have proved a large deviations principle for discrete-time processes with jump rates across a *straight* boundary, and Alanyali and Hajek [AH96] has proved one for continuos-time processes. The local rate function we use is a natural generalization of theirs.

Our main result about the overload probability performance is the following.

Theorem 2.1 *If $x^*(\delta_{qos}) > 1$ and $\lambda \geq x^*(\delta_{qos})$ then*

$$\lim_{T\to\infty} I^*(T) = \delta_{qos}$$

If $x^*(\delta_{qos}) = 1$ or $\lambda < x^*(\delta_{qos})$ then

$$\lim_{T\to\infty} I^*(T) \geq \delta_{qos}$$

The first case corresponds to the situation when the stable equilibrium is on the segment $z = g(x)$ of the boundary, whereas the second is when the equilibrium is on the segment $x = 1$ or in the interior of $\mathcal{S}_a$.

A large T means that the average call holding time is much longer than the time-scale of the fluctuation of the bit-rate of the sources, and thus implies a regime of separation of call and burst time-scales. Thus, for large T, the memoryless scheme satisfies the QOS requirement in terms of the exponential decay rate of the overload probability, and also performs asymptotically as well as the scheme which have perfect prior knowledge in terms of average utilization (from (5)). We can conclude that for large capacities and a separation of time-scales, the memoryless scheme is *asymptotically optimal.* This is somewhat surprising since the measurement-based scheme has no *a priori* knowledge about the sources and only uses information about the current state of the network to make admission decisions. Due to space limitation, we shall only outline the intuition behind the proof of this result.

If the statistics of the bandwidth requirements of the call were known, then the maximum number of calls allowed in the system, approximately $nx^*(\delta_{qos})$, can be determined, such that the overload probability when there are that many calls in the system satisfy the QOS requirements. The measurement-based scheme essentially tries to estimate $nx^*(\delta_{qos})$ based on observing the current state of the network. Due to statistical fluctuations in the bandwidth requirements of the calls, the estimate may be too high due to atypically large number of calls being off. This may result in the control scheme making a mistake in accepting a new call which it should have rejected. However, the key point is that for a large system, the overload probability will deteriorate significantly only if the number of calls in the system is considerably larger than $nx^*(\delta_{qos})$; thus, making an occasional mistake in admitting a call is not fatal. Rather, making a succession of mistakes in accepting calls will lead to serious degradation in performance. However, because of the separation of the time-scales of call arrivals and fluctuations of bandwidth requirements, it is very unlikely for the state of the network to remain atypical over the period of time for the arrivals of a succession of calls. In fact, it can be shown that the probability of making a succession of mistakes is much smaller than that of overload due to having a typical number $nx^*(\delta_{qos})$ of calls in the system but atypically large number of calls being on.

This intuition can be translated into a proof of Theorem 2.1 via an analysis of the costs of the paths in F (the set of paths leading to overload, as defined in (7)) as T grows large. Two candidate paths are depicted in Fig. 1. Suppose the stable equilibrium is $(x^*(\delta_{qos}), px^*(\delta_{qos}))$ on the segment $z = g(x)$ of the boundary. Consider the straight-line path $P_1 = \{(x^*(\delta_{qos}), z) : z \in [px^*, 1]\}$. This corresponds to the event that the measurement-based scheme estimates the maximum admissible number of calls $nx^*(\delta_{qos})$ correctly, and overload occurs during call rejection phase with $nx^*(\delta_{qos})$ calls in the system but an atypical fraction of them turning on. Also, no call departs the system during this phase. We observe that as the average call holding time increases, the cost of this path approaches δ_{qos}, since the probability of overload when there are $nx^*(\delta_{qos})$ calls *permanently* in the system is approximately $\exp(-n\delta_{qos})$.

Now let us consider alternate paths of which an example is P_2, shown in Fig. 1. These paths consists of two phases. In the first phase, calls are admitted by mistake due to an atypically large fraction of the existing calls in the system turning off. This phase ends at time τ_1, resulting in nx_1 calls in the system, where $x_1 > x^*(\delta_{qos})$. In the second phase, new calls are rejected, but many of the existing calls turn back on, resulting in

overload. Let us consider the costs of these paths in the regime of large holding time T. First, observe that the call arrival rate is $\frac{\lambda}{T} \cdot n$; hence, for those paths with $\tau_1 << T$, the cost is high because it is unlikely to have $n(x_1 - x^*(\delta_{qos}))$ arrivals in duration τ_1. On the other hand, for those paths with long duration τ_1, the cost is also high because it is unlikely for the calls in the system to be in an atypical state for so long. Using similar logic, it can be shown that the costs of all paths bounded away from P_1 go to infinity uniformly as average holding time $T \to \infty$. This implies that the cost of the cheapest path approaches δ_{qos} as $T \to \infty$.

We have only considered the case when the equilibria is on the segment $z = g(x)$. For the cases when the stable equilibrium point is on the segment $x = 1$ or in the interior of $\mathcal{S}_a$, one can show that the cost of the cheapest path must in fact be greater than δ_{qos}.

So far, we have focused on only on-off sources for ease and concreteness of exposition. The analysis and the Theorem 2.1 carry over to the general case of Markov fluid sources with general state space.

References

[AH96] M. Alanyali and B. Hajek On large deviations of Markov processes with discontinuous statistics. submitted to *Annals of Applied Probability*, 1996.

[Cru91] R.L. Cruz. A calculus for network delay, part i: Network elements in isolation. *IEEE Transactions on Information Theory*, 37(1):114–141, January 1991.

[DEed] P. Depuis and R.S. Ellis. *A Weak Convergence Approach to the Theory of Large Deviations.* to be published.

[CK] C. Courcoubetis et. al. Admission control and routing in ATM networks using inferences from measured buffer occupancy. *to be published.*

[GKK95] R.J. Gibbens, F.P. Kelly, and P.B. Key. A decision-theoretic approach to call admission control in ATM networks. *IEEE Journal on Selected Areas of Communications*, pages 1101–1114, August 1995.

[Hui88] J.Y. Hui. Resource allocation for broadband networks. *IEEE Journal on Selected Areas of Communications*, December 1988.

[HW] I. Hsu and J. Walrand. Dynamic bandwidth allocation for ATM switches. *submitted to J. of Applied Probability.*

[SS91] H. Saito and K. Shiomoto. Dynamic call admission control in ATM networks. *IEEE Journal on Selected Areas of Communications*, 9:982–989, 1991.

[SW95] A. Schwarz and A. Weiss. *Large Deviations for Performance Analysis.* Chapman and Hall, 1995.

A Pricing Model for Networks with Priorities*

Hong Jiang, Ikhlaq Sidhu & Scott Jordan
hongjiang@lucent.com, isidhu@usr.com & scott@eecs.nwu.edu
Bell Laboratories, U. S. Robotics & Northwestern University

Abstract

In this paper, a centralized resource allocation model based on priority schemes is first proposed for best-effort networks in which traffic with all priorities are fully multiplexed. Then a priority pricing scheme is used to distribute the resource allocation process. Users and the network exchange their characterizations on QoS, resource demand and prices via contract negotiation. The contract negotiations take place among three network layers: users, priority classes and the network, in a distributed and dynamic fashion.

1 Introduction

Resource allocation and pricing for high speed networks have attracted much research interests [1] [2] [3] [5] [6] [9] [10] [11] [12] in recent years. The connection establishment process described in [5] [6] relies on resource reservation as a mechanism to guarantee desired QoS of each connection. This type of resource management demands significant effort in user traffic characterization and admission control. Most importantly, the degree of statistical multiplexing is only limited to virtual circuits within a virtual path since further multiplexing requires a more complex resource management procedure. These drawbacks of resource reservation lead to a very different type of resource management often used by networks that provide best-effort services, in which the networks resort to other mechanisms to enhance network performance and provide diverse QoS: Priority schemes are a prime example of this. The trade-offs between resource reservation and priority schemes are: Best-effort networks with priorities do not guarantee the QoS, even though a network could use priorities to guarantee the QoS by controlling the amount of traffic admitted into each priority level. The characterization or the measurement of various QoS in terms of user demand could be challenging for a priority network; however, the advantage is that traffic sources are fully multiplexed among all priority classes.

In this paper, we adopt a connection establishment process using priorities to manage network resources. The connection establishment process consists of two separate stages: user and network characterizations, and then contract negotiation. Finally, a centralized mathematical model to maximize network economic efficiency is formulated, and the methods using pricing to distribute this maximization problem are outlined.

1.1 User Characterization

A user's objective here is to maximize his consumer surplus, which is equal to the benefit minus the cost of a service by choosing the appropriate demand for each priority class.

Similarly, a user's benefit of a service is defined in terms of demand and the QoS. In existing models [5] [6], it is assumed that users do not have the flexibility to choose a circuit bundle (or QoS). For instance, video

* This work was completed at Northwestern University.

users will use the video circuit bundles and voice users will only use the voice circuit bundles. The main argument for this assumption is that a video user usually will not choose a voice circuit bundle to send his video stream because there might be a mismatch between the voice circuit bundle's QoS and the desired QoS of his video application.

However, in a best-effort network with priorities, the QoS of the priority classes are not guaranteed and are varying in time, depending on user demand. A user can choose any one of the M priority classes, or he can mix different priority classes to transmit his traffic stream. For instance, a video user may compress a video stream at two levels, and he might want to use two priority classes to transmit the stream. In case of network congestion, the lower priority packets will be dropped first. However, for non-real-time services, if each data packet is equally important, then the user might only choose one out of M priority classes to send his data stream.

By incorporating user demand cross elasticity[1], better economic efficiency could be achieved. For example, when the QoS of a priority class becomes unsatisfactory to a user, he may decide to send fewer packets to this priority class and more packets to others. Hence, user benefit functions are defined as $ben(QoS_1, \lambda_{i1}, QoS_2, \lambda_{i2}, ...QoS_M, \lambda_{iM})$ to reflect the demand cross elasticity across priority classes, where λ_{ij} is user i's demand of priority class j. The definition of the benefit function shows that a user's benefit of a service depends on both the demand and the QoS of all priorities.

Consider a network with two priority classes, a user benefit function could look like the one in Figure 1 with QoS_1 and QoS_2 fixed. Certainly such a definition of benefit function is very broad and ideal, and could be difficult for a user to obtain. In practice, a user can choose a simplified version of the benefit function, or even a discrete benefit function as an approximation.

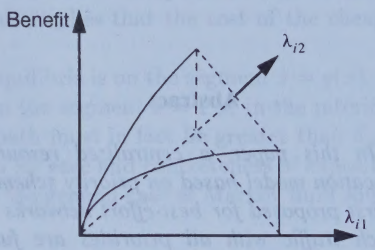


Figure 1 Benefit vs. demands

1.2 Network Characterization

The network's objective is to maximize its total user benefit by optimally allocating resources among priority classes and by choosing the best combination of QoS.

The network charges users on a per packet basis. Thus the total charge to a user for a connection would be $\sum_j P_j \lambda_{ij}$, where P_j is the price charged for priority j packets. The caveat of this pricing base has been discussed in [13] since some users with bursty traffic streams might not be charged enough to offset the negative externalities, such as prolonged delay and additional packet loss. Therefore, there might be some loss of network efficiency here. However, the benefit of complete sharing among all users might well compensate for the loss.

In high-speed networks, multiple services with diverse QoS require that the networks also provide diverse QoS. QoS in a best-effort network is often defined by mean delay and loss probability. Voice applications are very delay-sensitive but relatively loss-insensitive, and the data applications such as e-mail are loss-sensitive but relatively delay-insensitive. Hence, the networks need to decide a priority policy that can best meet users' QoS needs among many

1. Cross elasticity here means that users have the flexibility to choose among all priority classes.

priority policies. There has been vast literature on designing priority policies for computer networks [1] [4] [7] [14].

After the network determines a priority policy, it needs to communicate the policy to its users so that they can take advantage of such a policy and determine their demand for each priority class. On the network's side, given user demand of each priority class, the network needs to characterize the QoS of each priority class and its sensitivity in response to user demand. If the QoS and its sensitivity can be measured by the network, then the network can control the QoS and admission by measurement. Thus the tasks of user traffic characterization can be greatly reduced.

2 Contract Negotiation

A contract negotiation process is a distributed resource allocation and efficiency maximization process. Pricing is important in giving users incentives to choose the right QoS and demand for both himself and the network. Just imagine, if all four priority classes have the same price, then every user will choose the class with the best QoS, rendering the priority policy useless.

In the contract negotiation, the network must first decide a priority policy and inform its users of the policy so that users can take full advantage of the priorities by tailoring their demand accordingly. Second, the network and the users reach an agreement through an iterative process. The network first sets the prices for all priority classes, and declares the projected QoS of each class. The users respond with their demand to each priority class based on the information given by the network. The network then computes the QoS given the aggregate demand to each priority class, and announces a new set of prices and QoS. This iterative process continues until the projected QoS truly matches the QoS provided to the users. As user benefit functions and demand change over time, the negotiation takes place at a granularity comparable to the change of users' characteristics.

Before mathematical models are established for the network maximization problem, the notations and the assumptions are stated as follows. Without loss of generality, four priority classes are still used in the model.

λ_{ij} : the arrival rate or demand of user i to priority class j.

$\lambda_j = \sum_i \lambda_{ij}$: the aggregate arrival rate or demand to priority class j.

D_l : the mean delay of priority class l projected and announced by the network.

L_k : the loss probability of priority class k projected and announced by the network.

$ben_i(D_1, L_1, D_2, L_2, D_3, L_3, D_4, L_4, \lambda_{i1}, \lambda_{i2}, \lambda_{i3}, \lambda_{i4}) = ben_i(\{D_l\}, \{L_k\}, \{\lambda_{ij}\})$: the benefit function of user i in terms of mean delay, loss probability, and demand to each priority class.

$B = \sum_i ben_i$: the aggregate benefit of all users.

$FD_l(\lambda_1, \lambda_2, \lambda_3, \lambda_4) = FD_l(\{\lambda_j\})$: the mean delay of priority class l as a function of demand to each priority class.

$FL_k(\lambda_1, \lambda_2, \lambda_3, \lambda_4) = FL_k(\{\lambda_j\})$: the mean loss probability of priority class k as a function of demand to each priority class.

α_l, β_k : the Lagrangian multipliers of constraints (2) and (3).

P_j : the price per packet charged by priority class j to its users.

Assume that the user benefit functions are differentiable and jointly concave in λ_{ij}, D_l and L_k, and that they are increasing in λ_{ij} and decreasing in D_l and L_k. Assume also that the mean delay functions FD_l and the mean loss probability functions FL_k are

differentiable, jointly convex and increasing in λ_j for all priority class j.

First, a centralized mathematical model is formulated, where only a single node is considered. Pricing is introduced to decompose the centralized model into a distributed model. Finally, the negotiation process based on the distributed model is outlined.

2.1 A Centralized Model

The network's objective is to maximize its total user benefit subject to network constraints.

$$\max_{D_l, L_k, \lambda_{ij}} ben_i(\{D_l\}, \{L_k\}, \{\lambda_{ij}\}) \tag{1}$$

subject to constraints:

$$D_l = FD_l(\{\lambda_j\}) \qquad \forall l \tag{2}$$

$$L_k = FL_k(\{\lambda_j\}) \qquad \forall k \tag{3}$$

$$D_l, L_k, \lambda_{ij} \geq 0 \qquad \forall (l, k, i, j) \tag{4}$$

The Lagrangian function of the maximization problem is:

$$G = \sum_i ben_i(\{D_l\}, \{L_k\}, \{\lambda_{ij}\}) + \sum_l \alpha_l (D_l - FD_l(\{\lambda_j\})) + \sum_k \beta_k (L_k - FL_k(\{\lambda_j\})) \tag{5}$$

Since the objective function is concave, and the delay and loss functions are convex, the Lagrangian function is jointly concave in D_l, L_k and λ_j. This is a concave program, Kuhn-Tucker conditions [8] ((6)-(14)) are sufficient and necessary for the global optimal solution.

$$\frac{\partial G}{\partial \lambda_{ij}} = \frac{\partial}{\partial \lambda_{ij}} ben_i - \sum_l \alpha_l \frac{\partial FD_l}{\partial \lambda_{ij}} - \sum_k \beta_k \frac{\partial FL_k}{\partial \lambda_{ij}} \leq 0$$
$$= \frac{\partial}{\partial \lambda_{ij}} ben_i - \sum_l \alpha_l \frac{\partial FD_l}{\partial \lambda_j} - \sum_k \beta_k \frac{\partial FL_k}{\partial \lambda_j} \qquad \forall (i, j) \tag{6}$$

$$\lambda_{ij} \left(\frac{\partial}{\partial \lambda_{ij}} ben_i - \sum_l \alpha_l \frac{\partial FD_l}{\partial \lambda_j} - \sum_k \beta_k \frac{\partial FL_k}{\partial \lambda_j} \right) = 0 \qquad \forall (i, j) \tag{7}$$

$$D_l - FD_l(\{\lambda_j\}) = 0 \qquad \forall l \tag{8}$$

$$L_k - FL_k(\{\lambda_j\}) = 0 \qquad \forall k \tag{9}$$

$$\frac{\partial G}{\partial D_l} = \sum_i \frac{\partial}{\partial D_l} ben_i + \alpha_l \leq 0 \qquad \forall l \tag{10}$$

$$D_l \left(\sum_i \frac{\partial}{\partial D_l} ben_i + \alpha_l \right) = 0 \qquad \forall l \tag{11}$$

$$\frac{\partial G}{\partial L_k} = \sum_i \frac{\partial}{\partial L_k} ben_i + \beta_k \leq 0 \qquad \forall k \tag{12}$$

$$L_k \left(\sum_i \frac{\partial}{\partial L_k} ben_i + \beta_k \right) = 0 \qquad \forall k \tag{13}$$

$$\lambda_{ij}, D_l, L_k \geq 0 \qquad \forall (l, k, i, j) \tag{14}$$

Note that out of Kuhn-Tucker conditions, there are $4I + 8 + 8$ number of equations (not including the complimentary slackness conditions and (14)), where I represents the total number of users using the network. There are also the same number of variables in these equations: D_l, L_k, λ_{ij}, α_l and β_k. Therefore, if a global optimal solution exists, then it can be found by simultaneously solving these equations. Solving such a problem could be computationally prohibitive for a moderately-sized network. Hence, we investigate how to decentralize the model in an attempt to obtain the same optimal total user benefit achieved by the centralized model.

2.2 A Distributed Model

In the distributed model in [5] [6], since the network is structured according to four layers (users, circuit bundles, virtual paths, and physical trunks), the negotiations naturally occur among these layers. Since currently only a single node is considered, the natural structure of network has only three layers: users, priority classes and the node. The analogy between the two models is that priority classes here are similar to the circuit bundles that differentiate themselves from others with QoS. The difference between priority classes and circuit bundles resides in the fact that the QoS is guaranteed and fixed for each circuit

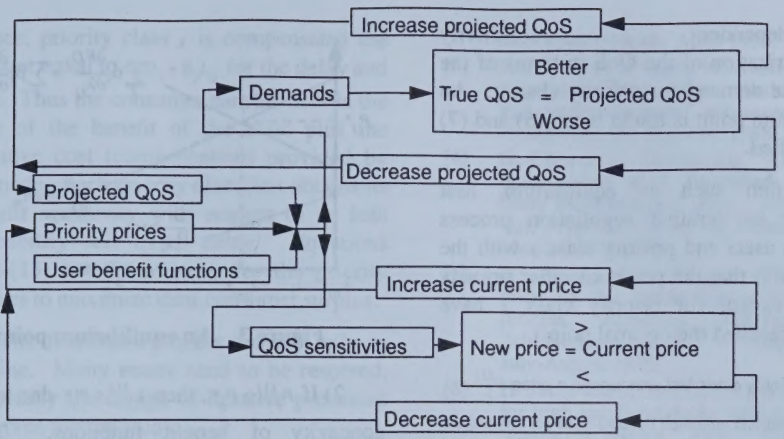


Figure2 Distributed contract negotiation for priority networks

bundle, while it is varying in time for a priority class.

A distributed contract negotiation could proceed in the fashion illustrated by Figure 2. Each user negotiates with each priority class for the amount of traffic he is transmitting to this priority class given the price of the priority class and the priority-level parameters. Each priority class negotiates with the node for the amount of traffic allocated by the node given the prices. These levels of negotiations could take place at different time scales, and the contract parameters such as QoS, resource demand and prices vary dynamically.

2.3 User-priority class negotiation:

For this level of negotiation, the parameters at a priority class level, such as the $QoS(D_l, L_k)$ and Lagrangian multipliers (α_l, β_k), are considered fixed. Users determine their demand to priority class j, and the priority class in turn sets the price given user demand.

User i maximizes his consumer surplus:

$$\max_{\lambda_{ij}} ben_i(\{D_l\}, \{L_k\}, \{\lambda_{ij}\}) - \sum_j P_j \lambda_{ij} \qquad (15)$$

subject to constraint: $\lambda_{ij} \geq 0$.

The optimal solution for maximal consumer surplus satisfies the following conditions:

$$\frac{\partial}{\partial \lambda_{ij}} ben_i - P_j \leq 0 \qquad \forall j \qquad (16)$$

$$\lambda_{ij}\left(\frac{\partial}{\partial \lambda_{ij}} ben_i - P_j\right) = 0 \qquad \forall j \qquad (17)$$

Equations (16) and (17) together indicate that if there exists a demand λ_{ij}^* for priority class j such that $\frac{\partial}{\partial \lambda_{ij}^*} ben_i - P_j = 0$, then λ_{ij}^* is the optimal demand for user i. If no such positive demand can be found to satisfy the equation, then the optimal demand $\lambda_{ij}^* = 0$.

Priority class j's strategy could be derived from equations (6) and (7). The equations suggest that if priority class j sets the price to be $P_j = \sum_l \alpha_l \frac{\partial FD_l}{\partial \lambda_j} + \sum_k \beta_k \frac{\partial FL_k}{\partial \lambda_j}$ given the fixed QoS (D_l, L_k) and the Lagrangian multipliers (α_l, β_k), then it can induce the users to behave in a socially optimal way. $\frac{\partial FD_l}{\partial \lambda_j}$ and $\frac{\partial FL_k}{\partial \lambda_j}$ are the

system-dependent sensitivity characterization of the QoS in terms of the aggregate demand for priority class j. An equilibrium point is found when (6) and (7) are satisfied.

To find such an equilibrium, first consider an iterative negotiation process between users and priority class j with the assumption that the prices of other priority classes except for priority class j have already reached the optimal points.

1. The priority class first announces a price $P_j^{(1)}$.
2. Users respond with their demand. The priority class then obtains the aggregate demand $\lambda_j^{(1)}$.
3. The priority class computes a new price $fP_j^{(1)} = \sum_l \alpha_l \frac{\partial FD_l}{\partial \lambda_j^{(1)}} + \sum_k \beta_k \frac{\partial FL_k}{\partial \lambda_j^{(1)}}$. If $fP_j^{(1)} = P_j^{(1)}$, then the equilibrium price P_j^* is found.
4. If $fP_j^{(1)} \neq P_j^{(1)}$, the priority class would choose a new price $P_j^{(2)}$ between the values of $P_j^{(1)}$ and $fP_j^{(1)}$. Then repeat steps 1, 2 and 3.

Theorem 1:

1) There exists an equilibrium price P_j^* for priority class j such that equations (6) and (7) are satisfied.

2) In the above iterative process, P_j^* is between the values of $P_j^{(1)}$ and $fP_j^{(1)}$.

Proof:

1) Due to the concavity and convexity of benefit function and QoS functions, $\frac{\partial}{\partial \lambda_{ij}} ben_i$ is decreasing and $\sum_l \alpha_l \frac{\partial FD_l}{\partial \lambda_j} + \sum_k \beta_k \frac{\partial FL_k}{\partial \lambda_j}$ is increasing since α_l and β_k are non-negative. Only two types of users are possible. Type n users can not afford the service, thus their demand $\lambda_j^* = 0$. Type m users choose their demand so that equation $\frac{\partial}{\partial \lambda_{ij}} ben_i - \sum_l \alpha_l \frac{\partial FD_l}{\partial \lambda_j} - \sum_k \beta_k \frac{\partial FL_k}{\partial \lambda_j} = 0$ can be satisfied (Figure 3). Therefore, there exists an equilibrium price point.

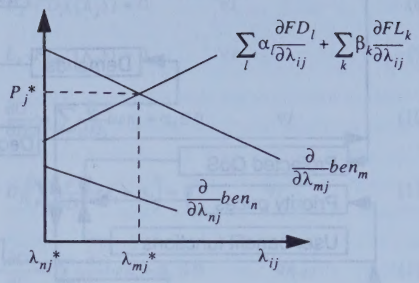


Figure 3 An equilibrium point

2) If $P_j^{(1)} \geq P_j^*$, then $\lambda_j^{(1)} \leq P_j^*$ due to the concavity of benefit functions. Thus $fP_j^{(1)} \geq P_j^*$ due to the convexity of $P_j = \sum_l \alpha_l \frac{\partial FD_l}{\partial \lambda_j} + \sum_k \beta_k \frac{\partial FL_k}{\partial \lambda_j}$. Hence $fP_j^{(1)} \leq P_j^* \leq P_j^{(1)}$. Similarly, if $P_j^{(1)} \leq P_j^*$, then $fP_j^{(1)} \geq P_j^* \geq P_j^{(1)}$. Q.E.D

Even though Theorem 1 does not guarantee that the iterative process will converge to the equilibrium price, it points out the right direction for the price setting.

Complexity arises when all four priorities are considered simultaneously during an iterative process. It is not clear how the iterative processes of all four classes should proceed. More research needs to be done to study the interactions and convergence among these priority classes.

2.4 Priority classes-the node negotiation:

The negotiation between the priority classes and the node results in the optimal solutions for class-level parameters, such as QoS and the prices charged to the priority classes by the node. The node's strategy could be derived from (8)-(9). The node maximizes its total user benefit subject to the delay and loss constraints ((8)-(9)). Priority classes maximize their consumer surplus. Note that in equations (10)-(13), Lagrangian multipliers (α_l, β_k) are positive.

Hence, priority class j is compensated the amount equal to $\alpha_j D_j + \beta_j L_j$ for the delay and loss. Thus the consumer surplus here is the sum of the benefit of the class plus the negative cost (compensation) provided by the node. Each priority class first obtains its benefit sensitivity with respect to its loss probability and mean delay. Equations (10)-(13) are the strategy for the priority classes to maximize their consumer surplus.

The distributed process above is only an outline. Many issues need to be resolved, especially the design of iterative processes deserves further study.

3 Conclusion

A connection establishment process is outlined for best-effort networks with priorities, where priority pricing is introduced to enhance network economic efficiency. Since the QoS of best-effort networks cannot be guaranteed, both the optimal QoS of and user demand for each priority class need to be determined by the network. It is shown that the optimal price charged by a priority class to its users equals the weighted sum of delay and loss sensitivities with respect to its demand. The optimal prices for delay and loss charged by the node to a priority class are negative and are equal to the sensitivity of the aggregate benefit of the priority class with respect to delay and loss respectively. Based on these optimal conditions for maximal network efficiency, a framework of a distributed negotiation process is outlined.

References

[1] A. Y. Lin and J. A. Silveste. "Priority queueing strategies and buffer allocation protocols for traffic control at an ATM integrated broadband switching system." *IEEE Journal on Selected Areas in Communications*, 9(9): 1524-1536, December, 1991.

[2] F. P. Kelly. "Tariffs, policing and admission control for multiservice networks." *10th UK Teletraffic Symposium*, BT Laboratories, Martlesham Heath, April 1993.

[3] F. P. Kelly. "Routing in circuit-switched networks: Optimization, shadow prices and decentralization." *Advanced Applied Probability*, 20: 112-144, 1988.

[4] G. Chen and I. Stavrakakis. "ATM traffic management with diversified loss and delay requirements." *Proc. Infocom'96*, Vol. 3, pp1037-1044, San Francisco, CA, March 24-28, 1996.

[5] H. Jiang and S. Jordan. "The role of price in the connection establishment process." *European Transactions on Telecommunications*, 6(4):421-429, July-August 1995.

[6] H. Jiang and S. Jordan. "A pricing model for high speed networks with guaranteed quality of service," *Proc. Infocom'96*, Vol. 2, pp888-895, San Francisco, CA, March 24-28, 1996.

[7] H. J. Chao and N. Uzun. "An ATM queue manager handling multiple delay and loss priorities." *IEEE/ACM Transactions on Networking*, 3(6):652-659, December 1995.

[8] J. Franklin. "Methods of mathematical economics: linear and nonlinear programming: fixed-point theorems." *New York: Springer-Verlag*, 1980.

[9] J. K. Mackie-Mason and H. R. Varian. "Pricing the Internet", *Second International Conference on Telecommunication Systems Modeling and Analysis*, pp378-393, Nashville, Tennessee, March 24-27, 1994.

[10] J. Murphy and L. Murphy. "Bandwidth allocation by pricing in ATM networks." preprint.

[11] R. Cocchi, D. Estrin, S. Shenker and L. Zhang. "Pricing in computer networks: motivation, formulation, and example." *IEEE/ACM Transactions on Networking*, 1(6):614-627, December 1993.

[12] S. H. Low and P. P. Varaiya. "A new approach to service provisioning in ATM networks." *IEEE Transactions on Networking*, 1(3):547-553, 1993.

[13] S. Jordan and H. Jiang. "Connection establishment in high speed networks." *IEEE Journal on Selected Areas in Communications*, 13(7): 1150-1161, September, 1995.

[14] Y. Takagi, S. Kino, and T. Takahashi. "Priority assignment control of ATM line buffers with multiple QoS classes." *IEEE Journal on Selected Areas in Communications*, 9(7): 1078-1091, September, 1991.

Asymptotically good sequences of curves and codes

Ruud Pellikaan *

October 1, 1996

1 Introduction

The parameters of a linear block code over the finite field $\mathbb{F}_q$ of *length* n, *dimension* k and *minimum distance* d will be denoted by $[n,k,d]_q$ or $[n,k,d]$. The quotient k/n is called the *information rate* and denoted by $R = k/n$ and the *relative minimum distance* d/n is denoted by δ.

A sequence of codes $(C_m | m \in \mathbb{N})$ with parameters $[n_m, k_m, d_m]$ over a fixed finite field $\mathbb{F}_q$ is called *asymptotically good* if n_m tends to infinity, and d_m/n_m tends to a non-zero constant δ, and k_m/n_m tends to a non-zero constant R for $m \to \infty$. Let $H_q(0) = 0$ and

$$H_q(x) = x\log_q(q-1) - x\log_q x - (1-x)\log_q(1-x)$$

for $0 < x \leq (q-1)/q$ be the entropy function. Then there exist asymptotically good sequences of codes attaining the the Gilbert-Varshamov (GV) bound

$$R \geq 1 - H_q(\delta)$$

It was shown by Tsfasman, Vlăduţ and Zink [23, 22] that there exist asymptotically good geometric Goppa codes on modular curves that satisfy the TVZ bound

$$\delta + R \geq 1 - \frac{1}{\sqrt{q}-1},$$

if q is a square. If moreover $q \geq 49$, then these codes are better than the GV bound. The theory of modular curves is very deep and the construction of these curves and their codes can be done in theory with polynomial complexity [19, 22] but are still too involved for an actual construction.

In this paper the latest results on the effective construction of asymptotically good codes and curves will be surveyed.

*Discrete Mathematics, Eindhoven University of Technology, P.O. Box 513, 5600 MB, Eindhoven, The Netherlands

2 Algebraic geometry codes

Consider a geometric object $\mathcal{X}$ with a subset $\mathcal{P}$ consisting of n points which are enumerated by $P_1, \ldots, P_n$. Suppose that we have a vector space L over $\mathbb{F}_q$ of functions on $\mathcal{X}$ with values in $\mathbb{F}_q$. Thus $f(P_i) \in \mathbb{F}_q$ for all i and $f \in L$. In this way one has an evaluation map

$$ev_{\mathcal{P}} : L \longrightarrow \mathbb{F}_q^n$$

which is defined by $ev_{\mathcal{P}}(f) = (f(P_1), \ldots, f(P_n))$. If this evaluation map is linear, then its image is a linear code.

The classical example of the above situation is given by *Reed-Solomon (RS)* codes. Here the geometric object $\mathcal{X}$ is the affine line over $\mathbb{F}_q$, the points are n distinct elements of $\mathbb{F}_q$ and L is the vector space of polynomials of degree at most $k-1$ and with coefficients in $\mathbb{F}_q$. This code has parameters $[n, k, n-k+1]$ if $k \leq n$. The length of a RS code is at most q.

If we take as geometric object $\mathcal{X}$ the *affine space* of dimension m over $\mathbb{F}_q$, for the set $\mathcal{P}$ all the q^m points of this affine space, and as vector space all polynomials of degree at most r, then we get the *Reed-Muller (RM)* codes of order r in m variables over $\mathbb{F}_q$. The length of these codes is not bounded, but they do not give a sequence of asymptotically good codes.

Let the geometric object be an (affine) *variety*, that is to say the zero set in affine space of some polynomials . Every variety has a *dimension* and a variety of dimension one is called an *algebraic curve*. If $\mathcal{X}$ is an algebraic curve over $\mathbb{F}_q$, $\mathcal{P}$ a set of n distinct points of $\mathcal{X}$ that are defined over $\mathbb{F}_q$, and L a vector space of rational functions with a certain prescribed behaviour of their poles and zeros, then we get the *geometric Goppa* codes also known as *algebraic geometry* (AG) codes. The parameters of these codes are determined by the theorem of *Riemann-Roch*, and they satisfy the following bound

$$k + d \geq n + 1 - g,$$

where g is an invariant of the curve called its *genus*.

3 Asymptotically good sequences of curves and codes

The information rate R and the relative minimum distance δ of an algebraic geometry code on a curve of genus g with n points that are defined over $\mathbb{F}_q$ satisfy

$$R + \delta \geq 1 - \frac{g-1}{n}.$$

In order to construct good codes we therefore need curves with low genus and many $\mathbb{F}_q$-rational points. For a curve over $\mathbb{F}_q$ of genus g with N $\mathbb{F}_q$-rational

points the Hasse-Weil bound says

$$N \leq q + 1 + 2g\sqrt{q}.$$

Let

$$A(q) = \limsup_{g \to \infty} \frac{N_q(g)}{g},$$

where $N_q(g)$ is the maximal number of $\mathbb{F}_q$-rational points on curves over $\mathbb{F}_q$ of genus g. The Hasse-Weil bound implies

$$A(q) \leq 2\sqrt{q}.$$

This has been improved to the Drinfeld-Vlăduţ (DV) bound [3]

$$A(q) \leq \sqrt{q} - 1 \tag{1}$$

When q is a square, Ihara [16]and Tsfasman, Vlăduţ, and Zink [23, 22] showed that

$$A(q) = \sqrt{q} - 1 \ ,$$

by studying the number of rational points on *modular curves* over finite fields. This in turn means that there exists a sequence of codes satisfying TVZ bound which is better than the GV bound when $q \geq 49$ in a certain range, and this fact was the starting point of the current interest in algebraic geometry codes.

Manin and Vlăduţ [19, 22] have shown that the construction of the modular curves and the corresponding codes can be done with polynomial complexity, of degree 20 for classical modular curves and degree 30 for Drinfeld modular curves. The degree for the latter has been reduced to 17 by Lopez [18]. But these constructions have no practical meaning. Others have tried to improve the complexity too [1, 2].

Apart from the fact that the construction has a high degree of complexity, the theory of modular curves is also deep and complex.

The paper of Justesen et al. [17] treats the construction and decoding of the class of codes on plane curves. This was the begining of an active period of research on decoding algorithms that culminated in the paper of Feng-Rao [4] on the majority voting among unknown syndromes. For a survey on this we refer to [15].

This history is mentioned because it gave the start of an elementary treatment of AG codes [5, 6, 8, 14] that is also known under the slogan:

"AG codes without AG".

Suppose that a decoding algorithm guarantees to find a unique closest code word for all received words with t errors, then the minimum distance is at least $2t + 1$. This observation lead to the Feng-Rao bound on the minimum distance, or more generally the order bound on the generalized Hamming weights [13].

4 The effective construction of asymptotically good sequences of codes

Many researchers have tried to construct asymptotically good sequences of curves in an elementary way and in a preprint [7] from 1994 by Feng and Rao the authors claimed to have found asymptotically good codes in an elementary way using what they called *generalized Klein curves* which are defined by the equations

$$X_i X_{i+1}^3 + X_{i+1} + X_i^3 = 0 \text{ for } i = 1, \ldots, m-1$$

over $\mathbb{F}_8$. Notice that the equations are of the following type

$$F(X_i, X_{i+1}) = 0 \text{ for } i = 1, \ldots, m-1,$$

where

$$F(X, Y) = XY^3 + Y + X^3.$$

This is the defining polynomial of the affine Klein quartic. This curve has the property that for every given nonzero element $x \in \mathbb{F}_q$ there are exactly 3 nonzero solutions in $\mathbb{F}_8$ of the equation $F(x, Y) = 0$ in Y. Therefore by induction this gives a curve with $7 \cdot 3^{m-1}$ points with nonzero coordinates in $\mathbb{F}_8$. Feng and Rao gave many more examples of polynomials $F(X, Y)$ over a finite field $\mathbb{F}_q$ of degree a in Y, and a subset S of $\mathbb{F}_q$, they called a set of *complete selected roots*, such that for any given $x \in$S there exist exactly a distinct $y \in S$ such that $F(x, y) = 0$. In this way they gave a lot of examples of curves with $|S| \cdot a^{m-1}$ rational points.

From the point of view of algebraic geometry one must compute the genus of these curves. A sequence of curves $(\mathcal{X}_m | m \in \mathbb{N})$ is called asymptotically good if $g(\mathcal{X}_m)$ tends to infinity and the following limit exists and

$$\lim_{m \to \infty} \frac{N(\mathcal{X}_m)}{g(\mathcal{X}_m)} > 0,$$

where $g(\mathcal{X})$ is the genus of $\mathcal{X}$ and $N(\mathcal{X})$ is the number of $\mathbb{F}_q$-rational points of $\mathcal{X}$. Frey, Rück and Stichtenoth [9] proved already some negative results in this direction. But this could not be applied to the generalized Klein curves. It was suggested to change the equations of the curves as follows

$$X_i X_{i+1}^2 + X_{i+1} + X_i^2 = 0 \text{ for } i = 1, \ldots, m-1$$

over $\mathbb{F}_4$. It turned out that this gave a tower of so-called Artin-Schreier extensions and Garcia and Stichtenoth [10] generalized it to

$$X_i^{r-1} X_{i+1}^r + X_{i+1} - X_i^r = 0 \text{ for } i = 1, \ldots, m-1,$$

where $q = r^2$, and calculated the genera and the number of $\mathbb{F}_q$-rational points. Thus they proved that the curves were asymptotically good and attain the DV bound.

Now the equations are of the form

$$F(X_i, X_{i+1}) = 0 \text{ for } i = 1, \ldots, m-1$$

with

$$F(X,Y) = X^{r-1}Y^r + Y - X^r.$$

The affine plane curve with equation $F(X,Y) = 0$ has the property that for every given nonzero element $x \in \mathbb{F}_q$ there are exactly r nonzero solutions in $\mathbb{F}_q$ of the equation $F(x,Y) = 0$ in Y. This is seen by multiplying the equation by X and replacing XY by Z. Then the equation $Z^r + Z = X^{r+1}$ is obtained, which defines the Hermitian curve over $\mathbb{F}_q$. Therefore the curve $\mathcal{C}_2$ has at least $(q-1)r$ points with nonzero coordinates in $\mathbb{F}_q$. Consider the map

$$\pi_m : \mathcal{C}_m \longrightarrow \mathcal{C}_{m-1}$$

defined as $\pi(x_1, \ldots, x_{m-1}, x_m) = (x_1, \ldots, x_{m-1})$. If $(x_1, \ldots, x_{m-1})$ is a given $\mathbb{F}_q$-rational point of $\mathcal{C}_{m-1}$ and $x_{m-1} \neq 0$, then there are exactly r possible nonzero values for $x_m \in \mathbb{F}_q$ such that $(x_1, \ldots, x_{m-1}, x_m)$ is a point of $\mathcal{C}_m$. Therefore by induction it is shown that

$$N(\mathcal{C}_m) \geq (q-1)r^{m-1}.$$

The genus of the curve $\mathcal{C}_m$ is computed by induction by applying the formula of Hurwitz-Zeuthen to the covering $\pi_m : \mathcal{C}_m \to \mathcal{C}_{m-1}$. In this case it turns out to be an Artin-Schreier covering [21]. It is easier to view this in terms of function fields. Let F_m be the function field of $\mathcal{C}_m$. Then $F_1 = \mathbb{F}_q(Z_1)$ and F_m is obtained from F_{m-1} by adjoining a new element Z_m that satisfies the equation

$$Z_m^r + Z_m = X_{m-1}^{r+1},$$

where $X_{m-1} = Z_{m-1}/X_{m-2} \in F_{m-1}$ for $m \geq 2$, and $X_1 = Z_1$, $X_0 = 1$. The genus g_m of the curve $\mathcal{C}_m$, or equivalently of the function field F_m is equal to

$$g_m = \begin{cases} r^m + r^{m-1} - r^{\frac{m+1}{2}} - 2r^{\frac{m-1}{2}} + 1 & \text{if } m \text{ is odd}, \\ r^m + r^{m-1} - \frac{1}{2}r^{\frac{m+2}{2}} - \frac{3}{2}r^{\frac{m}{2}} - r^{\frac{m-2}{2}} + 1 & \text{if } m \text{ is even}. \end{cases}$$

See [10]. Thus the DV bound is attained. Later it was shown that the tower of the generalized Klein curves is asymptotically bad [11]. This does not mean that the codes on these curves as defined in [7] are not asymptotically good. This has not been decided yet. Joint work with de Boer seems to indicate that they are not asymptotically good. The reason for this lies in the fact that a lot of functions are missing if we restrict ourselves to polynomials. One has to consider *rational functions*.

5 The quest of the missing functions

It turns out that finding bases for the vector spaces involved in the construction of AG codes is difficult. This last part remains to be done in order to make

the codes really constructive. A first step in this direction is made by Voss and Høholdt in [24] for the curve $\mathcal{C}_3$.

If the functions in L have only poles at P, an $\mathbb{F}_q$-rational point not equal to $P_1, \ldots, P_n$, then we can associate to f its pole order at P which we denote by $\rho_P(f)$ or $\rho(f)$. The opposite of this function $-\rho_P(f)$ is equal to the discrete valuation of f at P. Let R be the ring of all rational functions on the curve $\mathcal{X}$ with poles only possibly at P. Then

$$\rho : R \longrightarrow \mathbb{N}_0 \cup \{-\infty\}$$

is a function that satisfies the following porperties:

(0) $\rho(f) = -\infty$ if and only if $f = 0$
(1) $\rho(\lambda f) = \rho(f)$ for all nonzero $\lambda \in \mathbb{F}$
(2) $\rho(f+g) \leq \max\{\rho(f), \rho(g)\}$
and equality holds when $\rho(f) < \rho(g)$.
(3) $\rho(fg) = \rho(f) + \rho(g)$
(4) If $\rho(f) = \rho(g)$, then there exists a nonzero $\lambda \in \mathbb{F}$ such that $\rho(f - \lambda g) < \rho(g)$.

for all $f, g, h \in R$. Here $-\infty + n = -\infty$ for all $n \in \mathbb{N}_0$. And is called a *weight function.*

Let $(\rho_l | l \in \mathbb{N})$ be the seqeunce of all nonnegative weights that appear in increasing order. Then $\rho_i < \rho_{i+1}$ for all i, and for all nonzero f there exists an i such that $\rho(f) = \rho_i$.

In this way we get an increasing sequence of codes

$$E(l) = \{ev_P(f) | f \in R, \rho(f) \leq \rho_l\}$$

and a decreasing sequence of codes $C(l) = E(l)^{\perp}$.

The set $S = \{\rho_i | i \in \mathbb{N}\}$ forms a *semigroup* . That is to say that $0 \in S$, and if $x, y \in S$, then $x + y \in S$. The parameters of the codes $E(l)$ and $C(l)$ can be determined in terms of properties of this semigroup. The elements of $\mathbb{N}_0 \setminus S$ are called *gaps* and the *number of gaps* plays the same role as the genus.

The proof of the existence of a weight function in an elementary way works quite well for so-called plane curves of type I [5, 6] and their generalizations [20].

Consider the $\mathbb{F}_{64}$-algebra

$$R = \mathbb{F}_{64}[X, Y]/(X^5 + Y^4 + Y).$$

Assume that R has a weight function ρ. Let x and y be the cosets in R of X and Y, respectively. Then $x^5 = y^4 + y$. Now $y \notin \mathbb{F}$, so $\rho(y) > 0$. and $\rho(y^4) = 4\rho(y) > \rho(y)$ by (5). Thus $\rho(y^4 + y) = \rho(y^4)$ by (2). Therefore

$$5\rho(x) = \rho(x^5) = \rho(y^4 + y) = 4\rho(y)$$

Thus the only possible solution is $\rho(x) = 4$ and $\rho(y) = 5$. One can show that there exists a weight function. See [14, 20].

Let $R = \mathbb{F}_4[X, Y]/(XY^2 + Y + X^2)$. By the same reasoning as in the example above that $\rho(x) = 2$ and $\rho(y) = 1$ if there exists a weight function ρ on R. But one can show that there exists no weight funktion on R. The reason for this is that the second example has two points at infinity.

Let F be the field of fractions of R, so it is the function field of the curve. If we allowe negative values for the weight function, then we get a weight function ρ_P for every point P of the curve and $v_P = -\rho_P$ is the *discrete valuation* at P. To every rational function is associated its *principal divisor* (f) defined as

$$(f) = \sum_P v_P(f)P.$$

Consider the following ring

$$R = \mathbb{F}_4[X_1, \ldots, X_m]/(X_i X_{i+1}^2 + X_i^2 + X_{i+1}, i = 1, 2, ..., m-1).$$

The coset of X_i in R is denoted by x_i. Now the principal divisors of the functions

$$x_m^{e_m} \cdots x_1^{e_1}.$$

will be investigated. Those monomials are tabulated that have only a pole at Q_m for $m = 2, 3$ and 4.

$m = 2$:						$m = 3$:						
e_2	e_1	Q_0	Q_1	Q_2		e_3	e_2	e_1	Q_0	Q_1	Q_2	Q_3
0	0	0	0	0		0	0	0	0	0	0	0
0	1	1	1	-2		0	0	1	1	1	2	-4
1	1	3	0	-3		0	1	1	3	3	0	-6
0	2	2	2	-4		0	0	2	2	2	4	-8
.	.	.	.	.		1	0	2	6	0	3	-9
0	i	i	i	-2i		0	1	2	4	4	2	-10
1	i	i+2	i-1	-2i-1		1	1	2	8	2	1	-11
.	.	.	.	.		0	0	3	3	3	6	-12
⋮	⋮	⋮	⋮	⋮		⋮	⋮	⋮	⋮	⋮	⋮	⋮

So the second table says for instance that:

$$(x_3 x_2 x_1^2) = 8Q_0 + 2Q_1 + Q_2 - 11Q_3.$$

If $m = 2$, then the genus is 1 and we have in the list one gap at 1. If $m = 3$, then the genus is 5 and we have in the list 5 gaps at $1, 2, 3, 5$ and 7.

$m = 4$:

e_4	e_3	e_2	e_1	Q_0	Q_1	Q_2	Q_3	Q_4
0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	2	4	-8
0	0	1	1	3	3	6	0	-12
0	0	0	2	2	2	4	8	-16
0	1	0	2	6	6	0	6	-18
0	0	1	2	4	4	8	4	-20
1	0	1	2	12	0	6	3	-21
0	1	1	2	8	8	4	2	-22
0	0	0	3	3	3	6	12	-24
0	1	0	3	7	7	2	10	-26
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮

In fact Q_2 is the sum of two points. The genus is 13, but now there are 16 gaps at 1,2,3,4,5,6,7,9,10,11,13,14,15,17,19 and 25. We have to find three more functions. Leonard found the functions

$$X_3(1 + X_1X_4 + X_1X_2), \quad \frac{X_1X_3(1 + X_1X_4)}{X_2}, \quad \text{and } X_1X_3(1 + X_1X_4)$$

that have weights 14, 15 and 19, respectively. Notice that monomials are not sufficient anymore, even if we allow negative exponents. With the help of AXIOM, a computer algebra package, Haché [12] found many more missing functions for $m > 4$.

A new tower of function fields T_m is given in [11] over $\mathbb{F}_q$ with $q = r^2$, where $T_1 = \mathbb{F}_q(X_1)$ and T_m is obtained from T_{m-1} by adjoining a new variable X_m that satisfies the equation:

$$X_m^r + X_m = \frac{X_{m-1}^r}{X_{m-1}^{r-1} + 1}.$$

This sequence of function fields attains the DV bound too.

In joint work with Stichtenoth and Torres we are able to give the missing functions for $m \leq 5$. To give some idea the following function is shown

$$\frac{X_3^2}{X_1} + \frac{\left(X_4 + \frac{X_3^2}{X_2}\right)^2}{X_3} + X_5.$$

References

[1] M. Bronstein, M. Hassner, A. Velasquez and C.J. Williamson, "Computer algebra algorithms for the construction of error correcting codes on algebraic curves," preprint 1991.

[2] W. Burge, M. Hassner, S. Watt and C.J. Williamson, "An efficient AXIOM representation of elliptic modular curves," preprint 1993.

[3] V.G. Drinfeld and S.G. Vlăduţ, "Number of points of an algebraic curve," *Func. Anal.*, vol. 17, pp. 53-54, 1983.

[4] G.-L. Feng and T.R.N. Rao, "Decoding of algebraic geometric codes up to the designed minimum distance," *IEEE Trans. Inform. Theory*, vol. IT-39, pp. 37-45, Jan. 1993.

[5] G.-L. Feng and T.R.N. Rao, "A simple approach for construction of algebraic-geometric codes from affine plane curves," *IEEE Trans. Inform. Theory*, vol. IT-40, pp. 1003-1012, July 1994.

[6] G.-L. Feng and T.R.N. Rao, "Improved geometric Goppa codes," Part I: Basic Theory, *IEEE Trans. Inform. Theory*, Nov. 1995, pp. 1678-1693.

[7] G.-L. Feng and T.R.N. Rao, Improved geometric Goppa codes, Part II, Generalized Klein codes, peprint 1994.

[8] G.-L. Feng, V. Wei, T.R.N. Rao and K.K. Tzeng, "Simplified understanding and efficient decoding of a class of algebraic-geometric codes," *IEEE Trans. Inform. Theory*, vol. IT-40, pp. 981-1002, July 1994.

[9] G. Frey, G. Rück and H. Stichtenoth, "On the different of Abelian extensions of global fields," In Coding Theory and Algebraic Geometry, Luminy 1991, *Springer Lect. Notes Math.*, vol. 1518, pp. 26-32, 1992.

[10] A. Garcia and H. Stichtenoth, "A tower of Artin-Schreier extensions of function fields attaining the Drinfeld-Vlăduţ bound", *Invent. Math.*, vol. 121, pp. 211-222, 1995.

[11] A. Garcia and H. Stichtenoth, "On the asymptotic behaviour of some towers of function fields over finite fields," to appear in *Journ. Number Theory.*

[12] G. Haché, *Ph.D. Thesis,* INRIA, Univ. Paris VI, Sept. 1996.

[13] P. Heijnen and R. Pellikaan, "Generalized Hamming weights of q-ary Reed-Muller codes," preprint June 1996.

[14] T. Høholdt, J.H. van Lint and R. Pellikaan, "Algebraic geometry codes," preprint 1996.

[15] T. Høholdt and R. Pellikaan, "On the Decoding of Algebraic-Geometric Codes," *IEEE Trans. Inform. Theory*, vol. 41, pp. 1589-1614, nov. 1995.

[16] Y. Ihara, "Some remarks on the number of rational points of algebraic curves of finite fields," *Journ. Fac. Sc. Univ. Tokyo* IA, vol. 28, pp. 721-724, 1981.

[17] J. Justesen, K.J. Larsen, H. Elbrønd Jensen, A. Havemose and T. Høholdt, "Construction and decoding of a class of algebraic geometric codes," *IEEE Trans. Inform. Theory*, vol. IT-35, pp. 811-821, July 1989.

[18] B. López Jiménez, "Plane models of Drinfeld modular curves," *Ph.D. Thesis,* Unver. Complutense, Madrid, March 1996.

[19] Yu.I. Manin and S.G. Vlăduţ, "Linear codes and modular curves," *Journ. Sov. Math.* vol. 30, pp.2611-2643, 1985.

[20] R. Pellikaan, "On the existence of order functions," In *Proc. 2nd Shanghai Conference on Designs, Codes and Finite Geometry* 1996.

[21] H. Stichtenoth, *Algebraic function fields and codes*, Universitext, Springer-Verlag, Berlin 1993.

[22] M.A. Tsfasman and S.G. Vlăduţ, *Algebraic-geometric codes*, Mathematics and its Applications vol. 58, Kluwer Acad. Publ., Dordrecht 1991.

[23] M.A. Tsfasman, S.G. Vlăduţ and T. Zink, "Modular curves, Shimura curves and Goppa codes, better than Varshamov-Gilbert bound," *Math. Nachrichten*, vol. 109, pp. 21-28, 1982.

[24] C. Voss and T. Høholdt, "An explicit construction of a sequence of codes attaining the Tsfasman-Vlăduţ-Zink bound. The first steps," to appear in *IEEE Trans. Inform. Theory.*

The Weight Hierarchies and Chain Condition of a Class of Codes from Varieties over Finite Fields *

Xinwen Wu, Gui-Liang Feng, T. R. N. Rao

The Center for Advanced Computer Studies
University of Southwestern Louisiana
Lafayette, LA 70504, USA

Abstract

The generalized Hamming weights of linear codes were first introduced by Wei, which are fundamental parameters related to the minimal overlap structures of the subcodes and very useful in several fields. It was found that the chain condition of a linear code is convenient in studying the generalized Hamming weights of the product codes. In this paper we consider a class of codes defined over some varieties in projective spaces over finite fields, whose generalized Hamming weights can be determined by studying the orbits of subspaces of the projective spaces under the actions of classical groups over finite fields, i.e., the symplectic groups, the unitary groups and orthogonall groups. We will give the weight hierarchies of the codes from Hermitian varieties and prove that the codes satisfy the chain condition.

Index Terms: weight hierarchy, chain condition, codes from varieties, classical groups

1 Introduction

The generalized Hamming weights of a linear code were introduced by V. K. Wei [1]. Let $\mathbb{F}_q$ be a finite field, where q is a prime power. For any code D of block length n over $\mathbb{F}_q$, define the *support* $\chi(D)$ by

$$\chi(D) = \{i \mid c_i \neq 0 \text{ for some } (c_1, c_2, \cdots, c_n) \in D\},$$

and the *support weight* $w_s(D)$ by

$$w_s(D) = |\chi(D)|.$$

Let C be a linear $[n, k]$ code over $\mathbb{F}_q$. For any r, where $1 \leq r \leq k$, the *r-th generalized Hamming weight* of C is defined as

$$d_r(C) = \min\{\, w_s(D) \mid D \text{ is an } r-\text{dimensional subcode of } C\}.$$

Obviously, the minimum Hamming weight (minimum distance) of C is just $d_1(C)$. The *weight hierarchy* of C is then defined to be the set of generalized Hamming weights

$$\{d_1(C),\ d_2(C),\ \cdots, d_k(C)\}.$$

The following properties of the generalized Hamming weights are known for a q-ary $[n,\ k]$ code C.

*This work was supported in part by the National Science Foundation under Grant NCR-9505619, Louisiana Education Quality Support Fund under Grant LEQSF-(1994-96)-RD-A-36, and NASA Project NAG-W-4013.

(1) (Monotonicity) $1 \leq d_1(C) < d_2(C) < \cdots < d_k(C) \leq n$.

(2) (Duality) Let $C^{\perp}$ be the dual code of C. Then

$$\{d_r(C^{\perp}) | 1 \leq r \leq n-k\} = \{1, 2, \cdots, n\} \backslash \{n+1-d_r(C) | 1 \leq r \leq k\}.$$

(3) (The Griesmer-Wei bound)

$$d_r(C) \geq \sum_{i=0}^{r-1} \lceil \frac{d_1(C)}{q^i} \rceil, \quad \text{for } r = 1, 2, \cdots, k.$$

where $\lceil x \rceil$ denotes the smallest integer not less than x.

Many applications of generalized Hamming weights are known. They are useful in cryptography [1][2], in trellis coding [3][4], and in truncating a linear block code [5], etc..

The support weight distribution for irreducible cyclic codes was introduced by Helleseth, Kløve, and Mykkeltveit [10]. The generalized Hamming weights are actually the minimum support weights.

We say that a linear $[n, k]$ code C satisfies the *chain condition*, if there exist r-dimensional subcode D_r of C for $1 \leq r \leq k$ such that

$$w_s(D_r) = d_r(C), \quad r = 1, 2, \cdots, k.$$

and

$$D_1 \subset D_2 \subset \cdots \subset D_k.$$

It was found that the chain condition of a linear code is convenient in studying the generalized Hamming weights of the product codes. The following proposition can be found in [11],

Proposition 1 *If C satisfies the chain condition, so does its dual code $C^{\perp}$.*

2 The codes from varieties

We denote by $\mathbb{F}_q^{(k)}$ (resp. $PG(k-1, \mathbb{F}_q)$) the k dimensional vector space (resp. $(k-1)$ dimensional projective space) over $\mathbb{F}_q$. A q-ary linear $[n, k]$ code C is called a *projective code* if the columns of a generator matrix G of C can be regarded as distinct points of $PG(k-1, \mathbb{F}_q)$. So we have a point set in $PG(k-1, \mathbb{F}_q)$, whose elements are the column vectors of G, which will be denoted by $\mathcal{S}_G$ and called the point set arising from C via G. Different encoding matrices of C give rise to point sets which are projectively equivalent.

Two projective $[n, k]$ codes over $\mathbb{F}_q$ are said to be *equivalent*, if one can be obtained from the other by permuting the coordinates of the codewords and multiplying them by non-zero elements of $\mathbb{F}_q$.

Let G be a generator matrix of a projective $[n, k]$ code C and C' be a projective code equivalent to C. Then the same transformation which transforms C to C' will transform the encoding matrix G of C to an encoding matrix G' of C'. If G and G' are encoding matrices of two equivalent projective $[n, k]$ codes C and C', then $\mathcal{S}_G$ and $\mathcal{S}_{G'}$ are projectively equivalent.

Let us consider the finite field $\mathbb{F}_{q^2}$ with q^2 elements, where q is a power of prime. $\mathbb{F}_{q^2}$ has an involutive automorphism

$$a \longmapsto \overline{a} = a^q.$$

The fixed field of this automorphism is $\mathbb{F}_q$.

Let $k = \nu + l$, where $\nu > 0$, $l \geq 0$, and

$$I_{(\nu,l)} = \begin{pmatrix} I^{(\nu)} & \\ & 0^{(l)} \end{pmatrix}.$$

The set of points ${}^t(x_1, x_2, \cdots, x_k)$ satisfying

$$(x_1, \cdots, x_k) I_{(\nu,l)} \overline{{}^t(x_1, \cdots, x_k)} = 0$$

is a Hermitian variety in $PG(k-1, \mathbb{F}_{q^2})$, when $l = 0$, it is a nondegenerate Hemitian variety, and when $l > 0$, it is a degenerate Hermitian variety. We denote this Hermitian variety by $I_{(\nu,l)}$. Let $n = |I_{(\nu,l)}|$ be the number of points lying on $I_{(\nu,l)}$ in $PG(k-1, \mathbb{F}_{q^2})$. Then from [14], we have

$$\begin{aligned} n &= \frac{(q^\nu - (-1)^\nu)(q^{\nu-1} - (-1)^{\nu-1})}{q^2-1} q^{2l} + \frac{q^{2l}-1}{q^2-1} \\ &= \frac{q^{2\nu+2l-1} + (-1)^{\nu-1} q^{\nu+2l-1} + (-1)^\nu q^{\nu+2l} - 1}{q^2-1}. \end{aligned} \tag{1}$$

For each point of $I_{(\nu,l)}$, choose a system of coordinates and regard it as a k-dimensional column vector. Arrange these n column vectors in any order into $k \times n$ matrix, denote it also by $I_{(\nu,l)}$. It can be proved that $I_{(\nu,l)}$ is of rank k. Hence $I_{(\nu,l)}$ can be regarded as a generator matrix of a q^2-ary projective $[n,k]$-code, which will be denoted by $C_{(\nu,l)}$. Obviously,

$$S_{I_{(\nu,l)}} = I_{(\nu,l)}.$$

Remark 1 From (1), n is an increasing function of ν, and when $\nu \longrightarrow \infty$ we have $n \longrightarrow \infty$. For any integer ν, by the above discussion, we have an $[n_\nu, k_\nu]$ code $C_{(\nu,l)}$, where n_ν is given by (1) and $k_\nu = \nu + l$. So take $\nu = 1, 2, \cdots$, we get a sequence of $[n_\nu, k_\nu]$ long codes $\{C_{(\nu,l)}\}_{\nu=1}^{\infty}$.

As in [10], let G be a generator matrix of C, and for any column vector $x \in \mathbb{F}_q^{(k)}$, the k-dimensional column vector space over $\mathbb{F}_q$, let $m_G(x)$, the number of occurrences of the vector x as columns of G. Obviously, $w_s(C) = n - m_G(0^{(k)})$, where $0^{(k)}$ is the zero vector of $\mathbb{F}_q^{(k)}$. Let U be a subspace of $\mathbb{F}_q^{(k)}$ and define

$$m_G(U) = \sum_{x \in U} m_G(x).$$

If M is an $r \times k$ matrix of rank r, then MG generate an r-dimensional subcode of C, and any r-dimensional subcode is obtained in this way.

Let D be an r-dimensional subcode of C and MG be a generator matrix of D, where M is an $r \times k$ matrix of rank r. Define the *dual* of D to be

$$D^\perp = \{x \in \mathbb{F}_q^{(k)} | \, Mx = 0\}, \tag{2}$$

Then $D^\perp$ is an $(k-r)$-dimensional subspace. And for any r, where $1 \leq r \leq k$, the map $D \longmapsto D^\perp$ defined by (2) is a bijection from the set of r-dimensional subcodes of C to the set of $(k-r)$-dimensional subspaces of $\mathbb{F}_q^{(k)}$. We have the following lemmas.

Lemma 1 *Let D be a subcode of C. Then*

$$w_s(D) = n - m_G(D^\perp).$$

Lemma 2 *Let C be a q-ary projective $[n,k]$ code, G be a generator matrix of C, and $\mathcal{S}_G$ be the point set in $PG(k-1, \mathbb{F}_q)$ arising from C via G. For any r, where $1 \leq r \leq k$, let D_r be an r-dimensional subcode, then there is an $(k-r-1)$-flat of $PG(k-1, \mathbb{F}_q)$, P_{k-r-1}, such that*

$$w_s(D_r) = n - |P_{k-r-1} \cap \mathcal{S}_G|,$$

and

$$d_r(C) = n - \max\{|P_{k-r-1} \cap \mathcal{S}_G|\},$$

where P_{k-r-1} *runs through all* $(k-r-1)$*-flats of* $PG(k-1, \mathbb{F}_q)$.

Proof: It follows from Lemma 1. □

Wan [8] is the first to use the theory of geometry of classical groups over finite fields to study the generalized Hamming weights of the codes from varieties, he determined the generalized Hamming weights of the codes from nondegenerate quadrics and proved that the codes satisfies the chain condition. Wan and Wu [9] determined the weight hierarchies and generalized weight spectra of the codes from degenerate quadrics by using the geometry of orthogonal groups. In this paper we will give the weight hierarchies of the codes $C_{(\nu,l)}$ and prove that $C_{(\nu,l)}$ satisfies the chain condition, for any positive integer ν, by using the geometry of unitary groups.

3 The geometry of unitary groups

Denote by $GL_{\nu+l}(\mathbb{F}_{q^2})$ the general linear group of degree $\nu+l$ over $\mathbb{F}_{q^2}$. Let

$$U_{\nu+l,\nu}(\mathbb{F}_{q^2}) = \{T \mid T \in GL_{\nu+l}(\mathbb{F}_{q^2}) \text{ and } {}^tTI_{(\nu,l)}\overline{T} = I_{(\nu,l)}\}.$$

Then $U_{\nu+l,\nu}(\mathbb{F}_{q^2})$ forms a group with respect to matrix multiplication, call the *singular unitary group* of degree $\nu+l$ and rank ν over $\mathbb{F}_{q^2}$.

Let $\mathbb{F}_{q^2}^{(\nu+l)}$ be the $(\nu+l)$-dimensional space of column vectors over $\mathbb{F}_{q^2}$. Clearly, $\mathbb{F}_{q^2}^{(\nu+l)}$ admits an action of $U_{\nu+l,\nu}(\mathbb{F}_{q^2})$ defined as follows:

$$\begin{array}{ccc} U_{\nu+l,\nu}(\mathbb{F}_{q^2}) \times \mathbb{F}_{q^2}^{(\nu+l)} & \longrightarrow & \mathbb{F}_{q^2}^{(\nu+l)} \\ (T, {}^t(x_1, \cdots, x_{\nu+l})) & \longmapsto & T\,{}^t(x_1, \cdots, x_{\nu+l}). \end{array}$$

This action induces an action on $PG(k-1, \mathbb{F}_{q^2})$, and the set of flats of $PG(k-1, \mathbb{F}_{q^2})$ is subdivided into orbits under $U_{\nu+l,\nu}(\mathbb{F}_{q^2})$. Let P be an $(m-1)$-flat, then it corresponds to an m-dimensional subspace of $\mathbb{F}_{q^2}^{(k)}$, and there is a $k \times m$ matrix whose m columns form a basis of this subspace, if no ambiguity aries, we denote all of them by P.

Let P be an m-dimensional subspace of $\mathbb{F}_{q^2}^{(k)}$. P and its corresponding $(m-1)$-flat are said to be *type* (m, s) if ${}^tPI_{(\nu,l)}\overline{P}$ is cogredient to

$$M(m, s) = \begin{pmatrix} I^{(s)} & \\ & 0^{(m-s)} \end{pmatrix}.$$

Denote the set of subspaces or flats of type (m, s) by $\mathcal{M}(m, s; \nu+l, \nu)$ and let

$$N(m, s; \nu+l, \nu) = |\mathcal{M}(m, s; \nu+l, \nu)|.$$

Let E be the subspace of $\mathbb{F}_{q^2}^{(\nu+l)}$ generated by $e_{\nu+1}, \cdots, e_{\nu+l}$. An m-dimensional subspace P or an $(m-1)$-flat P is said to be *type* (m, s, t) if

(1) P is of type (m, s) and

(2) $\dim(P \cap E)$=t.

Denote the set of all subspaces or flats of type (m, s, t) by $\mathcal{M}(m, s, t; \nu+l, \nu)$ and let

$$N(m, s, t; \nu+l, \nu) = |\mathcal{M}(m, s, t; \nu+l, \nu)|.$$

The following Lemmas 3-6 will be used in the sequel and can be found in [14].

Lemma 3 *$\mathcal{M}(m,s,t;\nu+l,\nu)$ is non-empty if and only if*

$$t \le l \quad \text{and} \quad 2s \le 2(m-t) \le \nu + s. \tag{3}$$

Moreover, if $\mathcal{M}(m,s,t;\nu+l,\nu)$ is non-empty, then it is an orbit of subspaces or flats under $U_{\nu+l,\nu}(\mathbb{F}_{q^2})$.

Lemma 4 *$\mathcal{M}(m,s;\nu+l,\nu)$ is non-empty if and only if*

$$\max\{0, \frac{2m-\nu-s}{2}\} \le \min\{l, m-s\}. \tag{4}$$

Moreover, $\mathcal{M}(m,s;\nu+l,\nu)$ is the disjoint union of $\mathcal{M}(m,s,t;\nu+l,\nu)$ where t satisfies (3).

Lemma 5 *Suppose that (3) holds, then*

$$N(m,s,t;\nu+l,\nu) = q^{s(\nu+s-2m+2t)+2(m-t)(l-t)} \times \frac{\prod_{i=\nu+s-2m+2t+1}^{\nu}(q^i-(-1)^i)\prod_{i=l-t+1}^{l}(q^{2i}-1)}{\prod_{i=1}^{s}(q^i-(-1)^i)\prod_{i=1}^{m-s-t}(q^{2i}-1)\prod_{i=1}^{t}(q^{2i}-1)}.$$

Lemma 6 *Suppose that (4) holds, then*

$$N(m,s;\nu+l,\nu) = \sum_{t=\max\{0,[\frac{2m-\nu-s+1}{2}]\}}^{\min\{l,m-s\}} N(m,s,t;\nu+l,\nu).$$

Now we can prove the following lemmas.

Lemma 7 *Let $m \ge 1$ and P be an $(m-1)$-flat in $PG(k-1,\mathbb{F}_{q^2})$. Then*

$$|P \cap I_{(\nu,l)}| = |\,{}^tPI_{(\nu,l)}\overline{P}|.$$

Lemma 8 *Let $m \ge 1$, and P_1, P_2 be both $(m-1)$-flats of $PG(k-1,\mathbb{F}_{q^2})$, which belong to the same orbit under the action of $U_{\nu+l,\nu}(\mathbb{F}_{q^2})$. Then*

$$|{}^tP_1I_{(\nu,l)}\overline{P_1}| = |{}^tP_2I_{(\nu,l)}\overline{P_2}|.$$

Lemma 9 *Let $m \ge 1$ and P be an $(m-1)$-flat of type (m,s,t). Then*

$$|P \cap I_{(\nu,l)}| = \frac{q^{2m-1}+(-1)^sq^{2m-s}+(-1)^{s-1}q^{2m-s-1}-1}{q^2-1},$$

which is independent of t.

Lemma 10 *Let $m \ge 1$ and P be an $(m-1)$-flat of type (m,s). Then*

$$|P \cap I_{(\nu,l)}| = \frac{q^{2m-1}+(-1)^sq^{2m-s}+(-1)^{s-1}q^{2m-s-1}-1}{q^2-1}.$$

4 The weight hierarchies and chain condition

Theorem 1 *The generalized Hamming weights of the q^2-ary projective $[n,k]$ code $C_{(\nu,l)}$ is as follows.*

1) When ν is even,

$$d_r(C_{(\nu,l)}) = \begin{cases} \dfrac{q^{2\nu+2l-1}-q^{2(\nu+l-r)-1}}{q^2-1}, & \text{for } r=1,2,\cdots,\nu/2, \\ \dfrac{q^{2\nu+l-1}-q^{\nu+2l-1}+q^{\nu+2l}-q^{2(\nu+l-r)}}{q^2-1}, & \text{for } r=\nu/2,\cdots,\nu+l. \end{cases}$$

2) When ν is odd,

$$d_r(C_{(\nu,l)}) = \begin{cases} \dfrac{q^{2\nu+2l-1}-q^{\nu+2l}-q^{2(\nu+l-r)-1}+q^{\nu+2l-2}}{q^2-1}, & \text{for } r=1,\cdots,(\nu-1)/2, \\ \dfrac{q^{2\nu+l-1}-q^{\nu+2l-1}+q^{\nu+2l}-q^{2(\nu+l-r)}}{q^2-1}, & \text{for } r=(\nu+1)/2,\cdots,\nu+l. \end{cases}$$

□

Corollary 1 *Let ν be even and $r \le \nu$. Then $d_r(C_{(\nu,l)})$ meets the Griesmer-Wei bound.* □

Corollary 2 *The sequence of infinite linear codes $\{C_{(\nu,l)}\}_{\nu=1}^{\infty}$ has a nonzero asymptotically relative minimum distance.*

Proof: Consider $l = 0$. Let n_ν and d_ν be the code length and minimum distance of C_ν respectively, we have

$$n_\nu = \frac{q^{2\nu-1}-q^{\nu-1}+q^\nu-1}{q^2-1}, \quad \text{and} \quad d_\nu = q^{2\nu-3},$$

so,

$$\delta_\nu = \frac{d_\nu}{n_\nu} \longrightarrow \frac{q^2-1}{q^2} > 0, (\nu \longrightarrow \infty).$$

□

Theorem 2 *For any ν and l, codes $C_{(\nu,l)}$ satisfy the chain condition.*

Proof: Let P be an $(m-1)$-flat of type (m,s,t), where $m = k-r = \nu+l-r$. Then

$$|P \cap I_{(\nu,l)}| = \frac{q^{2m-1}+(-1)^s q^{2m-s}+(-1)^{s-1}q^{2m-s-1}-1}{q^2-1} = \frac{q^{2(\nu+l-r)-1}-1}{q^2-1} + \frac{q^{2(\nu+l-r)-1}}{q+1}(-\frac{1}{q})^s.$$

In $PG(k-1,\mathbb{F}_{q^2})$, there exist $(m-1)$-flats of type (m,s,t) if and only if

$$2s \le 2(m-t) \le \nu+s,$$

as $m = \nu+l-r$,

$$2s \le 2(\nu+l-r-t) \le \nu+s. \tag{5}$$

If ν is even. We consider the following two cases.

(1) When $2r \le \nu$, i.e., $r = 1,2,\cdots,\frac{\nu}{2}$, from (5), we have

$$s \ge \nu - 2r + 2(l-t).$$

Let $t = l$, (5) becomes $\nu - 2r \le s \le \nu - r$, so when P_{k-r-1} runs through all $(k-r-1)$-flats, $\max\{|P_{k-r-1} \cap I_{(\nu,l)}|\}$ is achieved by $(\nu+l-r-1)$-flats of type $(\nu+l-r, \nu-2r, l)$. Let $P_{\nu+l-2}$ be an $(\nu+l-2)$-flat of type $(\nu+l-1, \nu-2, l)$, then

$$d_1(C_{(\nu,l)}) = n - |P_{\nu+l-2} \cap I_{(\nu,l)}|.$$

Let $P_{\nu+l-3}$ be an $(\nu+l-3)$-flat of type $(\nu+l-2, \nu-4, l)$, then

$$d_2(C_{(\nu,l)}) = n - |P_{\nu+l-3} \cap I_{(\nu,l)}|.$$

And obviously we can assume that $P_{\nu+l-3} \subset P_{\nu+l-2}$. Proceeding in this way, we find a chain of flats $P_{\nu+l-4}$ of type $(\nu+l-4, \nu-6, l), \cdots$, $P_{\nu+l-i-1}$ of type $(\nu+l-i, \nu-2i, l)$, $i = 3, 4, \cdots, \frac{\nu}{2}$, with

$$d_i(C_{(\nu,l)}) = n - |P_{\nu+l-i-1} \cap I_{(\nu,l)}|,$$

and $P_{\frac{\nu}{2}+l-1} \subset P_{\frac{\nu}{2}+l} \subset \cdots \subset P_{\nu+l-3} \subset P_{\nu+l-2}$.

(2) When $2r > \nu$, i.e., $r = \frac{\nu}{2}+1, \cdots, \nu+l$, when $\frac{\nu}{2}+l-r \le t \le \nu+l-r$, we can take $s = 0$, and when P_{k-r-1} runs through all $(k-r-1)$-flats, $\max\{|P_{k-r-1} \cap I_{(\nu,l)}|\}$ is achieved by $(k-r-1)$-flats of type $(\nu+l-r, 0, t)$. Let $P_{\frac{\nu}{2}+l-2}$ be an $(\frac{\nu}{2}+l-1, 0, t)$ flat, where $l-1 \le t \le \frac{\nu}{2}+l-1$, then

$$d_{\frac{\nu}{2}+1}(C_{(\nu,l)}) = n - |P_{\frac{\nu}{2}+l-2} \cap I_{(\nu,l)}|.$$

And obviously we can assume $P_{\frac{\nu}{2}+l-2} \subset P_{\frac{\nu}{2}+l-1}$. Proceeding in this way, we can find a chain of flats

$$P_{-1} \subset P_0 \subset \cdots \subset P_{\frac{\nu}{2}+l-2} \subset P_{\frac{\nu}{2}+l-1}.$$

Combine (1) and (2), we get a chain of flats

$$P_{-1} \subset P_0 \subset \cdots \subset P_{\frac{\nu}{2}+l-2} \subset P_{\frac{\nu}{2}+l-1} \subset \cdots \subset P_{\nu+l-2},$$

where $P_{\nu+l-i-1}$ is a $(\nu+l-i-1)$-flat of type $(\nu+l-i, \nu-2i, l)$, when $i = 1, 2, \cdots, \frac{\nu}{2}$; $P_{\nu+l-i-1}$ is a $(\nu+l-i-1)$-flat of type $(\nu+l-i, 0, t)$, where $\frac{\nu}{2}+l-i \le t \le \nu+l-i$, when $i = \frac{\nu}{2}+1, \cdots, \nu+l-1$, respectively, and $P_{-1} = 0$, such that

$$d_i(C_{(\nu,l)}) = n - |P_{\nu+l-i-1} \cap I_{(\nu,l)}|,$$

for $i = 1, 2, \cdots, \nu+l$. Let

$$M_i = P^{\perp}_{\nu+l-i-1} = \{(x_1, \cdots, x_k) \in \mathbb{F}^{(k)}_{q^2} \mid \begin{matrix}(x_1, \cdots, x_k)\, {}^t(y_1, \cdots, y_k) = 0, \\ \forall\, (y_1, \cdots, y_k) \in P_{\nu+l-i-1}\end{matrix}\}.$$

Then M_i is an i-dimensional subspace of $\mathbb{F}^{(k)}_{q^2}$, and

$$M_1 \subset M_2 \subset \cdots \subset M_{\nu+l} = \mathbb{F}^{(k)}_{q^2}.$$

Denote also by M_i a matrix representation of the i-dimensional subspace M_i of $\mathbb{F}^{(k)}_{q^2}$, and let

$$D_i = M_i G,$$

where G is the generator matrix of code $C_{(\nu,l)}$. Then D_i is an i-dimensionl subcode of $C_{(\nu,l)}$ and by lemmas 2 and 3,

$$w_s(D_i) = n - |P_{\nu+l-i-1} \cap I_{(\nu,l)}|.$$

So we have a chain of subcodes

$$D_1 \subset D_2 \subset \cdots \subset D_{\nu+l} = C_{(\nu,l)},$$

such that

$$w_s(D_i) = d_i(C_{(\nu,l)}),$$

for $i = 1, 2, \cdots, \nu+l$. So when ν is even, $C_{(\nu,l)}$ satisfies the chain condition. If ν is odd, the proof is simlar, we omit the details. □

Theorem 3 *For any ν and l, the dual codes $C^{\perp}_{(\nu,l)}$ satisfy the chain condition.*

5 Conclusions

It is a challenging problem to determine the complete weight hierarchies of linear codes. Even for some classical codes, such as BCH codes, generalized Reed-Muller codes, etc., this problem is still unsolved. The chain condition is very useful in the study of weight hierarchies of product codes. We gave the complete weight hierarchies of the codes from Hermitian varieties and proved that the codes satisfy the chain condition.

References

[1] V. K. Wei, Generalized Hamming weights for linear codes, *IEEE Trans. Inform. Theory* **IT-37** (1991), 1412-1418.

[2] B. Chor, O. Goldreich, J.Hastad, J. Friedmann, S. Rudish, R. , The bit extraction problem ot t-resilient functions, *Proc. 26th Symp. Foundation Comput. Sci.*, (1985), 396-407.

[3] J. K. Wolf, Efficient Maximum Likelihood Decoding of Linear Block Codes Using a Trellis, *IEEE Trans. Inform. Theory* **IT-24**, (1978), 76-78.

[4] T. Kasami, T. Takata, T. Fujiwara, Shu Lin, On the Optimum Bit Orders with Respect to the State Complexity of Trellis Diagrams for Binary Linear Codes, *IEEE Trans. Inform. Theory* **IT-39**, (1993), 242-245.

[5] T. Helleseth, P. V. Kumar, The weight hierarchy of the Kasami codes, preprint.

[6] G. L. Feng, K. K. Tzeng, V. K. Wei, On the generalized Hamming weights of several classes of cyclic codes, *IEEE Trans. Inform. Theory* **IT-38** (1992), 1125-1130.

[7] J. W. P. Hirschfeld, M. A. Tsfasman, S. G. Vladut, The weight hierarchy of higher-dimensional Hermitian codes, *IEEE Trans. Inform. Theory* **IT-40** (1994), 275-278.

[8] Z. Wan, The weight hierarchies of the projective codes from nondegenerate Quadrices, *Designs, Codes and Cryptography*, **4**, 1994, 283-300.

[9] Z. Wan and Xinwen Wu, The weight hierarchies and generalized weight spectra of the projective codes from degenerate quadrics, to appear in *Discrete Mathematics.*

[10] T. Helleseth, T. Klϕve, J. Mykkeltveit, The weight distribution of irreducible cyclic codes with block length $n_1(\frac{(q^l-1)}{N})$, *Discrete Mathematics*, **18**(1977), 179-211.

[11] V. K. Wei and K. Yang, On the generalized Hamming weights of squaring construction and product codes, *IEEE Trans. Inform. Theory* **IT-39** (1993), 1709-1713.

[12] Xinwen Wu, Generalized Hamming weights of some classes of projective codes, Ph.D. Dissertation, Institute of Systems Science, Chinese Academy of Sciences, 1995.

[13] T. Helleseth, T. Klϕve, Φ. Ytrehus, Generalized Hamming Weights of Linear Codes, *IEEE Trans. Inform. Theory*, **38**(1992), 1133-1140.

[14] Z. Wan, Geometry of Classical Groups over Finite Fields, Studentlitteratur, Lund, 1993.

Construction and Decoding of a New Class of Double-Error-Correcting Codes from Hermitian Curves

Ba-Zhong Shen
Quantum Corporation
Shrewsbury, MA 01545

Kenneth K. Tzeng[1]
Department of Electrical
Engineering and Computer Science
Lehigh University, Bethlehem, PA

Abstract

A new method we introduced recently for constructing linear codes from algebraic curves is reviewed. The construction of a new class of double-error-correcting codes from Hermitian curves is presented in detail. These double-error-correcting codes are shown to have better code rate compared to the algebraic-geometric Hermitian codes of the same length. It is then shown that the decoding of these codes can be accomplished using two runs of the Berlekamp-Massey algorithm.

I Introduction

In an effort toward bridging the gap between cyclic codes and algebraic-geometric (AG) codes, we have recently introduced a new method for constructing linear codes from algebraic curves. Given a curve with N rational points and for an integer d^*, a code of length $n < N$ and minimum distance at least d^* can be constructed in a way resembling the construction of BCH and Reed-Solomon codes. In this paper, the construction of a new class of double-error-correcting codes from Hermitian curves is presented in detail. It is shown that, compared with AG codes of the same length from Hermitian curves, these new codes have higher code rate and potentially larger minimum distance. We then show that the decoding of these double-error-correcting codes can be accomplished using two runs of the Berlekamp-Massey algorithm and Forney's formula.

[1]This work was supported by the National Science Foundation under Grant NCR-9406043.

II New codes from Hermitian curves

Let $H(q)$ be the Hermitian curve over $\mathbf{F}_{q^2}$ defined by the equation

$$U^{q+1} + V^{q+1} + W^{q+1} = 0.$$

The function field of $H(q)$ is $\mathbf{F}_{q^2}(x, y)$, where the defining equation is

$$x^{q+1} = y^q + y.$$

This curve has $q^3 + 1$ rational points. Let α be a primitive element of $\mathbf{F}_{q^2}$. For $i = 0, \ldots, q^2 - 1$, denote

$$\alpha_0 = 0 \text{ and } \alpha_i = \alpha^{i-1}, \text{ for } i \geq 1,$$

and let $\beta_{i,j}$ denote the q solutions of $y^q + y = \alpha_i^{q+1}$ over $\mathbf{F}_{q^2}$. It is known that for any given i, $\beta_{i,1}, \beta_{i,2}, \ldots, \beta_{i,q}$ are distinct. Then the pair $(\alpha_i, \beta_{i,j})$ corresponds to a rational point of $H(q)$. We denoted this pair by $P_{i,j}$. Thus the set of all q^3 points $P_{i,j}$ corresponds the set of all the rational points of $H(q)$, except the point at infinity.

Using any $n \leq q^3$ rational points, we have shown in [1] that new codes $C(n, \delta)$ over $\mathbf{F}_{q^2}$ can be constructed having minimum distance $d \geq \delta + 1$. In the following, we will review the definition of the codes.

For any positive integer $n \leq q^3$, let $\theta \leq q^2 - 1$ and ξ be two non negative integers such that $n = \theta q^2 + \xi$. Thus, $\theta = \lfloor \frac{n}{q^2} \rfloor$. In the rest of this paper, any element $\boldsymbol{x}$ of $\mathbf{F}_{q^2}^n$ is always denoted as follows:

$$\boldsymbol{x} = \begin{cases} (\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_\theta) & \xi = 0 \\ (\underline{x}_1, \underline{x}_2, \ldots, \underline{x}_\theta, \underline{x}_{\theta+1}) & \xi \neq 0 \end{cases} \tag{1}$$

where

$$\underline{x}_i = (x_{0,i}, x_{1,i}, \ldots, x_{q^2-1,i}) \text{ for } 1 \leq i \leq \theta \text{ and } \underline{x}_{\theta+1} = (x_{0,\theta+1}, \ldots, x_{\xi-1,\theta+1}).$$

Given an integer $\delta > 0$, we define

$$R(\delta, v) := \begin{cases} \{0, 1, \ldots, \lfloor \frac{\delta}{v} \rfloor - 1\} & v \leq \delta \\ \emptyset & v > \delta \end{cases}.$$

Furthermore, define

$$\mathcal{R}(n, \delta) := \bigcup_{v=1}^{\lceil n/q^2 \rceil} \{(u, v-1) | u \in R(\delta, v)\}.$$

In particular, if $\delta = 4$ and $\lceil n/q^2 \rceil \geq 4$, we have

$$R(4,1) = \{0,1,2,3\}, R(4,2) = \{0,1,\}, R(4,3) = R(4,4) = \{0\}$$

and $\mathcal{R}(n, 4)$ consists of the following 8 elements

$$\begin{array}{llll} (0,0) & (1,0) & (2,0) & (3,0) \\ (0,1) & (1,1) & & \\ (0,2) & & & \\ (0,3) & & & \end{array}$$

A new code is then defined as follows.

Definition 1 A linear code $C(n,\delta)$ of length n is defined by

$$C(n,\delta) := \{\boldsymbol{c} \in \mathbf{F}_{q^2}^n \mid \sum_{j=1}^{\lceil n/q^2 \rceil} \sum_{i=0}^{m_j} c_{i,j}\alpha_i^\mu \beta_{i,j}^\nu = 0,\ (\mu,\nu) \in \mathcal{R}(n,\delta)\},$$

where

$$m_j = \begin{cases} q^2-1 & \text{if } j = 1,\ldots,\theta \\ \xi - 1 & \text{if } \xi \neq 0 \text{ and } j = \theta+1. \end{cases}$$

In particular, if $n = q^3$ then

$$C(q^3,\delta) := \{\boldsymbol{c} \in \mathbf{F}_{q^2}^{q^3} | \sum_{j=1}^{q} \sum_{i=0}^{q^2-1} c_{i,j}\alpha_i^\mu \beta_{i,j}^\nu = 0,\ (\mu,\nu) \in \mathcal{R}(n,\delta)\},$$

We call $\mathcal{R}(n,\delta)$ a *defining set* of the code.

The code $C(n,\delta)$ has minimum distance $d \geq \delta+1$ and dimension $k = n-\delta-\delta(n)$ when $\delta \leq q^2$, where $\delta(n) = |\mathcal{R}(n,\delta)| - \delta$.

When $n = q^2$, we have $\theta = 1$, $\xi = 0$ and $\mathcal{R}(q^2,\delta) = \{(0,0),\ldots,(\delta-1,0)\}$. Then the code

$$C(q^2,\delta) = \{\boldsymbol{c} \in \mathbf{F}_{q^2}^{q^2} | \sum_{i=0}^{q^2-1} c_{i,1}(\alpha_i)^j = 0,\ j = 0,1,\ldots,\delta-1\}$$

is an extended Reed-Solomon code.

Example Consider the Hermitian curve $H(4)$ over $\mathbf{F}_{16}$ with defining equation $x^5 = y^4 + y$. Let $P_{i,j} := (\alpha_i, \beta_{i,j})$ for $0 \leq i \leq 15$ and $1 \leq j \leq 4$ be its 64 rational points, where α is a primitive element of $\mathbf{F}_{16}$ with $\alpha^4 + \alpha + 1 = 0$. Let $n = 64$ and $\delta = 4$. Then

$$\mathcal{R}(64,4) = \{(0,0),(1,0),(2,0),(3,0),(0,1),(1,1),(0,2),(0,3)\}.$$

The code $C(64,4)$ is then defined as

$$\{(c_{0,1},c_{1,1},\ldots,c_{15,1},\ldots,c_{0,4},\ldots,c_{15,4}) \in \mathbf{F}_{4^2}^{64} | \sum_{i=0}^{15}\sum_{j=1}^{4} c_{i,j}\alpha_i^\mu \beta_{i,j}^\nu = 0, (\mu,\nu) \in \mathcal{R}(64,4)\}.$$

A parity check matrix of this code is

$$\begin{bmatrix}
1 & 1 & \ldots & 1 & 1 & \ldots & 1 & \ldots & 1 & \ldots & 1 \\
\alpha_0 & \alpha_1 & \ldots & \alpha_{15} & \alpha_0 & \ldots & \alpha_{15} & \ldots & \alpha_0 & \ldots & \alpha_{15} \\
\alpha_0^2 & \alpha_1^2 & \ldots & \alpha_{15}^2 & \alpha_0^2 & \ldots & \alpha_{15}^2 & \ldots & \alpha_0^2 & \ldots & \alpha_{15}^2 \\
\alpha_0^3 & \alpha_1^3 & \ldots & \alpha_{q^2-1}^3 & \alpha_0^3 & \ldots & \alpha_{15}^3 & \ldots & \alpha_0^3 & \ldots & \alpha_{15}^3 \\
\beta_{0,1} & \beta_{1,1} & \ldots & \beta_{15,1} & \beta_{0,2} & \ldots & \beta_{15,2} & \ldots & \beta_{0,4} & \ldots & \beta_{15,4} \\
\alpha_0\beta_{0,1} & \alpha_1\beta_{1,1} & \ldots & \alpha_{15}\beta_{15,1} & \alpha_0\beta_{0,2} & \ldots & \alpha_{15}\beta_{15,2} & \ldots & \alpha_0\beta_{0,4} & \ldots & \alpha_{15}\beta_{15,4} \\
\beta_{0,1}^2 & \beta_{1,1}^2 & \ldots & \beta_{15,1}^2 & \beta_{0,2}^2 & \ldots & \beta_{15,2}^2 & \ldots & \beta_{0,4}^2 & \ldots & \beta_{15,4}^2 \\
\beta_{0,1}^3 & \beta_{1,1}^3 & \ldots & \beta_{15,1}^3 & \beta_{0,2}^3 & \ldots & \beta_{15,2}^3 & \ldots & \beta_{0,4}^3 & \ldots & \beta_{15,4}^3
\end{bmatrix}$$

Let (n,k,d) denote a code with length n, dimension k and minimum distance d. Then the new code $C(64,4)$ is a $(64,56,\geq 5)$ codes while the best Hermitian codes (which are algebraic-geometric (AG)codes) on $H(4)$ of actual minimum distance 5 is a $(64,54,5)$ code. It is seen that codes from our construction have higher dimension. Moreover, we have the following general theorem

Theorem 1 *Consider Hermitiam curve $H(q)$ of genus g. Then whenever $\delta \leq g/(\ln q - E(q)$, the code $C(n,\delta)$ has the larger dimension than that of AG code on $H(q)$ with the same designed minimum distance $d^* = \delta + 1$, where*

$$E(l) := 1 - (\mathcal{E} + \frac{1}{2l} - \frac{1}{12l^2} + \frac{\epsilon_l}{120l^4}),\ 0 < \epsilon_l < 1$$

Since we can prove that

$$\frac{g}{\ln q - E(q)} \geq 4$$

for $q \geq 4$, we have the following corollary.

Corollary 1 *When $q \geq 4$, double-error correcting codes $C(n,4)$ have higher code rate than that of AG (Hermitian) code $C_\Omega(D, mP)$ of designed minimum distance 5.*

The above conclusion leads us to find a simple double-error-correcting algorithm. In the following, we listed parameters of double-error correction code $C(n,4)$ for $q = 3,4,5,7,8$. In the list, r denotes the number of check symbols.

q	n	d	r	Rate	
3	27∼19	5	7	.74∼.63	
	18∼14		6	.667∼.57	
	13∼10			.53∼.4	
	9		4	.556	Extended RS Code
4	64∼49	5	8	.875∼.837	
	48∼33		7	.854∼.788	
	32∼17		6	.813∼.647	
	16		4	.75	Extended RS Code
5	125∼76	5	8	.936∼.895	
	75∼51		7	.91∼.863	
	50∼26		6	.88 ∼.77	
	25		4	.84	Extended RS Code
7	343∼148	5	8	.977∼.946	
	147∼99		7	.953∼.93	
	98∼50		6	.939∼.88	
	49		4	.918	Extended RS Code
8	512∼193	5	8	.984∼.959	
	192∼133		7	.964∼.947	
	132∼65		6	.955∼.908	
	64		4	.938	Extended RS Code
q	$q^3 \sim 3q^2 + 1$	5	8		
	$3q^2 \sim 2q^2 + 1$		7		
	$2q^2 \sim q^2 + 1$		6		
	q^2		4		Extended RS Code

Remark From the above table, we found that many $C(n,4)$ codes are better than the extended RS codes.

III Decoding of double-error-correcting code $C(n,4)$

Since the construction of the code $C(n,\delta)$ is of the BCH and Reed-Solomon type, it is only natural that the decoding should more or less follow that of the decoding of BCH and RS codes. In this section, we will show that using the Berlekamp-Massey algorithm, a procedure can be designed for decoding all double-error-correcting codes $C(n,4)$.

We first briefly review the procedure for decoding a $(n, n-d+1, d)$ RS code over $\mathbf{F}_q$, where $n = q-1$. Let $R = \{1, \ldots, d-1\}$ be a defining set of the code. Then, for a certain primitive element α of $\mathbf{F}_q$, $\boldsymbol{c} = (c_0, \ldots, c_{n-1})$ is a codeword if and only if

$$\sum_{i=0}^{n-1} c_i \alpha^{ki} = 0 \quad \text{for } k = 1, \ldots, d-1.$$

Suppose $\boldsymbol{r} = \boldsymbol{c} + \boldsymbol{e}$ is a received vector, where $\boldsymbol{c} = (c_0, \ldots, c_{n-1})$ is the transmitted codeword and $\boldsymbol{e} = (e_0, \ldots, e_{n-1})$ is the error vector. For any integer k, define a syndrome

$$S_k := \sum_{i=0}^{n-1} e_i \alpha^{ki}.$$

Then, for $k = 1, \ldots, d-1$, $S_k = \sum_{i=0}^{n-1} r_i \alpha^{ki}$, can be computed. If the weight of the error vector $\mathrm{wt}(\boldsymbol{e}) \leq t = \lfloor (d-1)/2 \rfloor$, we have the following theorem which is a general version of well known Peterson decoding theorem [2].

Theorem 2 *Let $\alpha_1, \alpha_2, \ldots, \alpha_l$ be l distinct elements in $\mathbf{F}_q$. Let $\boldsymbol{e} \in \mathbf{F}_q^l$ be a vector with the weight $\mathrm{wt}(\boldsymbol{e}) \leq t$. For any integer k, define:*

$$S_k := \sum_{i=1}^{l} e_i \alpha_i^k.$$

Let $I = \{i \mid e_i \neq 0\}$. let $\sigma(x) = \sigma_\nu x^\nu + \sigma_{\nu-1} x^{\nu-1} + \cdots + \sigma_1 x + 1$ $(\nu \leq t)$ be a minimal degree polynomial satisfying

$$\begin{bmatrix} S_1 & S_2 & \cdots & S_{\nu+1} \\ S_2 & S_3 & \cdots & S_{\nu+2} \\ \vdots & \vdots & \ddots & \\ S_\nu & S_{\nu+1} & \cdots & S_{2\nu} \end{bmatrix} \begin{pmatrix} \sigma_\nu \\ \sigma_{\nu-1} \\ \vdots \\ \sigma_1 \\ 1 \end{pmatrix} = 0.$$

Then, the set of roots of $\sigma(x)$ is $\{\alpha_i^{-1} \mid i \in I\}$, namely $\sigma(x)$ is an locator polynomial of $\boldsymbol{e}$. Furthermore, $\boldsymbol{e}$ can be obtained from $\sigma(x)$ and $S_1, \ldots, S_{2\nu}$.

It is known that an efficient way to find $\sigma(x)$ is using the Berlekamp-Massey Algorithm. After obtaining the error locations, one can use Forney's formula to compute the error magnitudes.

Let $\boldsymbol{r} = \boldsymbol{c} + \boldsymbol{e}$ be a received vector, where $\boldsymbol{c} \in C(n,\delta)$ and $\boldsymbol{e}$ is an error vector. According to (1), we have

$$\boldsymbol{e} = (e_{0,1}, e_{1,1}, \ldots, e_{q^2-1,1}, e_{0,2}, \ldots, e_{q^2-1,2}, \ldots, e_{i,j}, \ldots).$$

For any integer pair (u, v) define

$$T_{u,v} := \sum_{j=1}^{\lceil n/q^2 \rceil} \sum_{i=0}^{m_j} e_{i,j} \alpha_i^u \beta_{i,j}^v,$$

where m_j is defined in Definition 1 and $(\alpha_i, \beta_{i,j})$ are solutions of $x^{q+1} = y^q + y$ over $\mathbf{F}_{q^2}$, which can be considered as the code locations. Then by the definition of $C(n, \delta)$, we have

$$T_{u,v} = \sum_{j=1}^{\lceil n/q^2 \rceil} \sum_{i=0}^{m_j} r_{i,j} \alpha_i^u \beta_{i,j}^v, \quad \text{for } (u, v) \in \mathcal{R}(n, \delta). \tag{2}$$

Let

$$E_i := \{j \mid e_{i,j} \neq 0\} \quad \text{for } 0 \leq i \leq q^2 - 1$$

and

$$I = \{i \mid E_i \neq \emptyset\}.$$

Then by defining

$$S_v^{(i)} = \sum_{j \in E_i} e_{i,j} \beta_{i,j}^v \quad \text{for } 0 \leq i \leq q^2 - 1 \text{ and } v \geq 0,$$

we have

$$T_{u,v} = \sum_{i \in I} S_v^{(i)} \alpha_i^u \tag{3}$$

Use these notation, we now develop an algorithm for decoding $C(n, 4)$ up to its designed minimum distance $d^* = 5$, i.e., to correct up to two errors. In the following, we always suppose that $n/q^2 > 3$, thus $\mathcal{R}(n, 4)$ always has 8 elements.

Since $\mathrm{wt}(\boldsymbol{e}) \leq 2 = (d^* - 1)/2$, we have $|I| \leq 2$. By (2), we know that $T_{u,0}$ can be computed from $\boldsymbol{r}$ for $u = 0, 1, 2, 3$. Suppose the $T_{u,0}$'s are not all zero. Then, since α_i, $i \in I$, are distinct, by Theorem 2, I and $S_0^{(i)}$ can be found from $T_{0,0}, T_{1,0}, T_{2,0}, T_{3,0}$ using the Berlekamp-Massey algorithm and Forney's formula. We then consider the following two cases,

i) $|I| = 1$. Denote $I = \{i\}$. Then we have

$$\sum_{j \in E_i} e_{i,j} \beta_{i,j}^v = S_v^{(i)} = T_{0,v} \quad \text{for } v = 0, 1, 2, 3.$$

Since $|E_i| \leq 2$ and the elements $\beta_{i,j}$ for $j \in E_i$ are distinct, then by Theorem 2 again, we can find E_i and $e_{i,j}$.

ii) $|I| = 2$. Denote $I = \{i_1, i_2\}$. We then have $|E_{i_1}| = |E_{i_2}| = 1$. Denote $E_{i_k} = \{j(i_k)\}$ for $k = 1, 2$. Then we have

$$e_{i_k, j(i_k)} = S_0^{(i_k)} \quad \text{for } k = 1, 2$$

The next step is to find the error locations. Since

$$S_1^{(i_1)} \alpha_{i_1}^u + S_1^{(i_2)} \alpha_{i_2}^u = T_{u,1}, \quad u = 0, 1. \tag{4}$$

and α_i, $i \in I$ are distinct, it is known that $S_1^{(i_1)}$ and $S_1^{(i_2)}$ can be obtained by solving the equations in (4). Finally, since $S_1^{(i_k)} = e_{i_k, j(i_k)} \beta_{i_k, j(i_k)}$, we can compute the error locations by

$$\beta_{i_k, j(i_k)} = S_1^{(i_k)} (S_0^{(i_k)})^{-1}, \quad k = 1, 2.$$

Then we have the following decoding algorithm.

Algorithm 1

Step 1 *Input the received vector* $\boldsymbol{r}$ *and compute* $T_{u,v}$ *from* $\boldsymbol{r}$ *for* $(u,v) \in \mathcal{R}(n,4)$;

Step 2 *Apply the Berlekamp-Massey Algorithm and Forney's formula to* $T_{0,0}, T_{1,0}, T_{2,0}$ *and* $T_{3,0}$ *to find* I *and* $S_0^{(i)}$;

Step 3 *If* $I = \emptyset$, *output* $\boldsymbol{c} = \boldsymbol{r}$ *and goto Step 6;*

Step 4 *i) If* $|I| = 1$, *say* $I = \{i\}$, *then apply the Berlekamp-Massey algorithm and Forney's formula to*

$$\sum_{j \in E_i} e_{i,j} \beta_{i,j}^v = T_{0,v} \quad \text{for } v = 0, 1, 2, 3,$$

to obtain E_i *and* $e_{i,j}$;

ii) If $|I| = 2$,

a) Compute $S_1^{(i)}$, $i \in I$ *by solving the following equations*

$$\sum_{i \in I} S_1^{(i)} \alpha_i^u = T_{u,1}, \qquad u = 0, 1;$$

b) For $i \in I$, *compute* $\beta_{i,j} = S_1^{(i)} (S_0^{(i)})^{-1}$ *which determines the element* j *in the sets* E_i;

c) Let $e_{i,j} = S_0^{(i)}$ *for* $j \in E_i$ *and* $i \in I$;

Step 5 *Output* $\boldsymbol{c}$, *where*

$$c_{i,j} = \begin{cases} r_{i,j} - e_{i,j} & \text{if } j \in E_i \text{ and } i \in I \\ r_{i,j} & \text{otherwise} \end{cases};$$

Step 6 *Stop.*

IV Conclusion

In [1], we have constructed a new class of codes from algebraic curves, in particular, Hermitian and generalized Hermitian curves. The construction is in a way resembling the construction of BCH and Reed-Solomon codes using a location set and a defining set. In this paper we consider double-error-correcting codes. It is shown that those codes have higher code rate than that of AG codes. Taking advantage of the construction of these codes, i.e., the defining set consists of two sequences of "consecutive elements", we have shown that the decoding of these codes can be accomplished using two runs of the Berlekamp-Massey algorithm and Forney's formula.

References

[1] B.-Z. Shen and K.K. Tzeng, "A New Construction of Codes from Algebraic Curves," in *Proceedings of 1995 IEEE International Symposium on Information Theory*, (Whistler, Canada), p. 98, Sept. 1995.

[2] W. Peterson and J. E.J. Weldon, *Error-Correcting Codes.* Cambridge, Mass. and Landon, England: The MIT. Press, 1972.

On the Classification of Self-Dual Codes

W. Cary Huffman
Department of Mathematical and Computer Sciences
Loyola University
Chicago, Illinois

Abstract

A number of authors have worked on the classification of extremal self-dual codes over the finite fields GF(2), GF(3), and GF(4). For certain small lengths a complete characterization is known, and for greater lengths, partial results have been obtained. A presentation of the current status of existence and characterization of these codes will be given. Included will be a brief description of the classification of the extremal Hermitian self-dual codes over GF(4) of lengths 18 and 20. Additionally, techniques have been developed to find codes with specified automorphisms. A description of these techniques and how they generalize to self-dual $\mathbb{Z}_4$ linear codes will also be presented.

1 Introduction

In this paper we consider linear codes, particularly extremal self-dual codes, over GF(q) with $q = 2$, 3, and 4. We will also consider linear codes over the Galois ring $\mathbb{Z}_4$.

An $[n, k, d]$ *linear code* $\mathcal{C}$ over $F_q = \mathrm{GF}(q)$ is a k-dimensional subspace of F_q^n of minimum nonzero weight d where the *weight* of a codeword $\mathbf{c} \in \mathcal{C}$ is the number of nonzero components of $\mathbf{c}$. Endow F_q^n with the ordinary inner product when $q = 2$ or 3 and with the Hermitian inner product when $q = 4$ $\langle \cdot, \cdot \rangle$ given by

$$\langle \mathbf{u}, \mathbf{v} \rangle = \sum_{i=1}^{n} u_i v_i \text{ when } q = 2 \text{ or } 3, \qquad \langle \mathbf{u}, \mathbf{v} \rangle = \sum_{i=1}^{n} u_i v_i^2 \text{ when } q = 4,$$

where $\mathbf{u} = u_1 u_2 \cdots u_n$ and $\mathbf{v} = v_1 v_2 \cdots v_n$ are in F_q^n. The *dual* of $\mathcal{C}$ is the $[n, n-k]$ code $\mathcal{C}^\perp$ defined by

$$\mathcal{C}^\perp = \{\mathbf{v} \in F_q^n \mid \langle \mathbf{u}, \mathbf{v} \rangle = 0 \text{ for all } \mathbf{u} \in \mathcal{C}\}.$$

A code $\mathcal{C}$ is *self-orthogonal* if $\mathcal{C} \subseteq \mathcal{C}^\perp$ and *self-dual* if $\mathcal{C} = \mathcal{C}^\perp$. Self-dual codes of length n have dimension $\frac{n}{2}$. A code $\mathcal{C}$ is *divisible* by Δ if all codeword weights are divisible by Δ. The following result, due to Ward [41], generalizes the Gleason–Pierce Theorem [1] and indicates why there is an interest in self-dual codes over F_2, F_3, and F_4.

Theorem 1.1 *Let $\mathcal{C}$ be a code over F_q (any q allowed) of length n and dimension $\frac{n}{2}$ that is divisible by $\Delta > 1$. Then either*

(i) *$q = 2$ and $\Delta = 2$, or*

(ii) *$q = 2$, $\Delta = 4$, and $\mathcal{C}$ is self-dual, or*

(iii) *$q = 3$, $\Delta = 3$, and $\mathcal{C}$ is self-dual, or*

(iv) *$q = 4$, $\Delta = 2$, and $\mathcal{C}$ is Hermitian self-dual, or*

(v) *$\Delta = 2$ and $\mathcal{C}$ is equivalent to the two-fold replication of $F_q^{n/2}$.*

Binary and ternary self-dual codes automatically satisfy Theorem 1.1 (i) and (iii) respectively. Binary self-dual codes satisfying (ii) are called *Type II* codes and must have lengths a multiple of 8. Binary self-dual codes satisfying (i) but not (ii) are called *Type I*. Codes satisfying (iii) and (iv) are called *Type III* and *Type IV* respectively. Codes of Type III must have lengths a multiple of 4. Codes of Types I and IV only need to be of even length.

The minimum weight of codes of Types I–IV are bounded according to the following result (see [9]):

Theorem 1.2 *Let C be an $[n, \frac{n}{2}, d]$ code of Type I, II, III, or IV. Then the following bounds hold for d:*

Type I: $d \leq 2\lfloor \frac{n}{8} \rfloor + 2$.

Type II: $d \leq 4\lfloor \frac{n}{24} \rfloor + 4$.

Type III: $d \leq 3\lfloor \frac{n}{12} \rfloor + 3$.

Type IV: $d \leq 2\lfloor \frac{n}{6} \rfloor + 2$.

Codes which meet the respective bounds in Theorem 1.2 are called *extremal*. In each case, extremal codes do not exist for sufficiently large n. In fact, the bound for Type I codes is met only for lengths $n = 2$, 4, 6, 12, 14, and 22 (see [41]). A stronger bound has been obtained for Type I codes in [7]; we will refer to codes that meet this bound as *optimal* Type I codes.

Theorem 1.3 *Let C be an $[n, \frac{n}{2}, d]$ Type I code. If $n \neq 2$, 12, 22, or 32 then $d \leq 2\lfloor \frac{n+6}{10} \rfloor$.*

We are obviously interested in finding codes only up to equivalence. We say that two codes are *equivalent* if one can be obtained from the other by permuting coordinates in the binary case, permuting and scaling coordinates in the ternary case, or permuting, scaling coordinates, and applying a Galois automorphism componentwise when the codes are over F_4.

2 Classification Techniques and Tables

Extremal codes of Types I–IV have been completely classified up to equivalence for some small lengths. The main technique, which we call *complete enumeration*, has been to find all codes, extremal or not, using 'gluing' theory. A 'mass formula' exists which tells how many total codes of a given type and length exist. A list of inequivalent codes together with their automorphism groups is produced. One computes the total number of codes equivalent to those in the list and then compares this count to that of the mass formula. If the count falls short, codes are added to the list until the count agrees with that of the mass formula. The extremal codes can then be picked out from all codes.

However, this technique appears to be infeasible for lengths greater than those already completed in this manner. For example, complete enumeration has been done for Type II codes of lengths 8, 16, 24, and 32. The next length 40 would require the classification of at least 17,000 inequivalent codes [3].

In Tables 1–4, we present the results of the classification where all extremal codes of given type have been entirely classified. In Table 1 the number of inequivalent Type

I extremal codes are given for the lengths $n = 2$, 4, 6, 12, 14, and 22; the number of inequivalent Type I optimal codes are given for the other lengths (marked with *). Those of lengths n with $2 \leq n \leq 20$ (see [30]) and $n = 22$ and 24 (see [33]) were classified using complete enumeration. Those of lengths $n = 26$, 28, and 30 were classified in [6, 32] as 'children' of the extremal Type II codes of length 32. The optimal Type I codes of length 32 were also found in [7] using these same extremal Type II codes of length 32 and arguments involving the 'shadow' code.

n	d	number
2	2	1
4	2	1
6	2	1
8*	2	1
10*	2	2
12	4	1

n	d	number
14	4	1
16*	4	1
18*	4	2
20*	4	7
22	6	1
24*	6	1

n	d	number
26*	6	1
28*	6	3
30*	6	13
32*	8	3

Table 1: Inequivalent extremal/optimal* Type I codes.

n	d	number
8	4	1
16	4	2
24	8	1
32	8	5

Table 2: Inequivalent extremal Type II codes.

Table 2 gives the number of extremal Type II codes of lengths up to 32. Using complete enumeration those of lengths 8 and 16 were presented in [30] and those of length 32 in [3]. The unique Type II code of length 24 is the extended Golay code, which follows from the more general result in [29, 39] that a linear (or nonlinear) code of length 24 and minimum distance 8 containing at least 2^{12} codewords, including $\mathbf{0}$ is equivalent to the extended Golay code.

n	d	number
4	3	1
8	3	1
12	6	1

n	d	number
16	6	1
20	6	6
24	9	2

Table 3: Inequivalent extremal Type III codes.

Table 3 shows the current status of the complete classification of the extremal Type III codes. Using complete enumeration those of lengths 4, 8, and 12 were classified in [28], those of length 16 in [5], and those of length 20 in [34]. Without using complete enumeration, but using instead the complete list of 59 inequivalent 24×24 Hadamard matrices, the extremal Type III codes of length 24 were found in [25].

The extremal Type IV codes of lengths 2 to 14 were classified using complete enumeration in [26] as were those of length 16 in [5]. Other techniques, described below,

n	d	number
2	2	1
4	2	1
6	4	1
8	4	1

n	d	number
10	4	2
12	6	0
14	6	1
16	6	4

n	d	number
18	8	1
20	8	2
24	10	0

Table 4: Inequivalent extremal Type IV codes.

were used to classify those of length 18 and 20 in [20] and length 24 in [24]. The results are summarized in Table 4. Length 22 is still open.

The technique for classifying the extremal Type IV codes $\mathcal{C}$ of lengths 20 and 24 is as follows. Let $\mathcal{B}$, respectively $\mathcal{D}$, be the largest subcode of $\mathcal{C}$ whose support is contained entirely in the left n_1, respectively right n_2, coordinates where $n_1 + n_2 = n$. Suppose $\mathcal{B}$ and $\mathcal{D}$ have dimensions k_1 and k_2, respectively. Then $\mathcal{C}$ has an $\frac{n}{2} \times n$ generator matrix

$$\text{gen}(\mathcal{C}) = \begin{bmatrix} B & O \\ O & D \\ E & F \end{bmatrix},$$

where $\text{gen}(\mathcal{B}) = [B\ O]$, $\text{gen}(\mathcal{D}) = [O\ D]$, and O is the appropriate size zero matrix. Let $\mathcal{B}^*$ be the $[n_1, k_1]$ code generated by B, $\mathcal{B}_E$ the code of length n_1 generated by the rows of B and E, $\mathcal{D}^*$ the $[n_2, k_2]$ code generated by D, and $\mathcal{D}_F$ the code of length n_2 generated by the rows of D and F. It can be shown that $\text{rank}(E) = \text{rank}(F) = \frac{1}{2}n - k_1 - k_2$, $k_2 = \frac{1}{2}n + k_1 - n_1$, $\mathcal{B}_E^{\perp} = \mathcal{B}^*$, and $\mathcal{D}_F^{\perp} = \mathcal{D}^*$. In each case choose $k_1 = 1$ and $n_1 = d$, the minimum weight, and order the coordinates so that the left n_1 coordinates support a minimum weight codeword whose nonzero coordinates make up the lone row of B. The above equations yield $k_2 = 3$ if $n = 20$ and $k_2 = 4$ if $n = 24$. If $n = 24$, the matrix D is unique up to equivalence and consequently F can be uniquely determined because $\mathcal{D}_F^{\perp} = \mathcal{D}^*$. It turns out that E cannot be completed when $\mathcal{C}$ is extremal. If $n = 20$, D can be reduced to 5 possibilities, each yielding a unique F. By a careful reordering of all 20 coordinates at various stages in the construction of possibilities for E, exactly two inequivalent extremal codes emerge.

The uniqueness of the $[18, 9, 8]$ extremal Type IV code $\mathcal{C}$ comes from a different argument. Let $\mathcal{C}_1$ be the $[16, 8, 6]$ code obtained from $\mathcal{C}$ by choosing all codewords of $\mathcal{C}$ which agree on the last two coordinates and deleting these two coordinates. Because of the agreement on these last two coordinates, $\mathcal{C}_1$ is Hermitian self-dual and hence extremal Type IV. Thus $\mathcal{C}_1$ is one of four possible codes from Table 4. Starting with a basis for each of these four codes consisting of weight 6 codewords, two equal nonzero components can be adjoined to each basis codeword to construct possibilities for 8 of the 9 basis vectors for $\mathcal{C}$. Using the known automorphism groups of $\mathcal{C}_1$ found in [5], we can reduce the number of possibilities. It turns out that only two of these extended bases generate $[18, 8, 8]$ codes $\mathcal{C}_2$. Inside each $\mathcal{C}_2^{\perp}$, there is a unique vector modulo $\mathcal{C}_2$, up to scalar multiplication and permuting the last two coordinates, which can possibly generate a minimum weight 8 code when added to the existing basis of $\mathcal{C}_2$. These possibilities in fact happen and the two codes have an automorphism of order 5. By [16] (see Table 8), the codes must both be equivalent to each other.

When complete enumeration seems impractical, it is reasonable to settle for less. For instance one could ask for all extremal codes of a given type which possess a particular automorphism. This has certainly been done in other circumstances. For example when

searching for the projective plane of order 10, initial attempts were made to find such a plane with an automorphism of a particular prime order. The first attempted classification of codes in this manner was presented by the author in [14], where it was shown that the only extremal Type II code of length 48 with a nontrivial odd order automorphism is the extended quadratic residue code (see Table 6). The following problem is still open.

Research Problem 2.1 *Are there other extremal Type II codes of length* 48*?*

The theory developed in [14] was generalized by Huffman and Yorgov in several papers including [16, 18, 42, 43]. Suppose a self-dual $[n, \frac{n}{2}, d]$ code $\mathcal{C}$ over F_q ($q = 2$, 3, or 4) has an automorphism σ of prime order p where $p \nmid q$ and $p \nmid (q-1)$. As rescaling columns does not change self-duality in the cases we are considering, $\mathcal{C}$ is equivalent to a self-dual $[n, \frac{n}{2}, d]$ code with a permutation automorphism of order p. Hence we may assume that σ is a permutation with c p-cycles $\Omega_1, \ldots, \Omega_c$ and $f = n - cp$ fixed points $\Omega_{c+1}, \ldots, \Omega_{c+f}$.

Let $\mathcal{R}_p = F_q[X]/(X^p - 1)$ denote the ring of polynomials over F_q modulo the ideal $(X^p - 1)$. Let $X^p - 1 = M_0(X)M_1(X)\cdots M_t(X)$ where $M_i(X)$ are irreducible over F_q for $0 \le i \le t$ with $M_0(X) = X - 1$. Then $\mathcal{R}_p$ is semisimple, and $\mathcal{R}_p = \mathcal{I}_0 \oplus \mathcal{I}_1 \oplus \cdots \oplus \mathcal{I}_t$ where $\mathcal{I}_j$ is the minimal ideal of $\mathcal{R}_p$ generated by $(X^p - 1)/M_j(X)$. Each $\mathcal{I}_j$ is in fact an extension field of F_q. In particular $\mathcal{I}_0 = \{k(1 + X + \cdots + X^{p-1}) \mid k \in F_q\} \simeq F_q$. Let r be the characteristic of F_q. Define the maps $\tau_{r^a,u} : \mathcal{R}_p \to \mathcal{R}_p$ for $0 \le a < p$ and $1 \le u < p$ by

$$\Big(\sum_{i \in \mathbb{Z}_p} a_i X^i\Big)\tau_{r^a,u} = \sum_{i \in \mathbb{Z}_p} a_i^{r^a} X^{ui}.$$

It is easy to see [16] that $\tau_{r^a,u}$ is a ring automorphism of $\mathcal{R}_p$, $\tau_{r^a,u} \in \mathrm{Gal}(\mathcal{I}_0)$, and that $\tau_{r^a,u}$ induces field isomorphisms as it permutes the fields $\mathcal{I}_1, \ldots, \mathcal{I}_t$. Finally define a form $(\cdot,\cdot)_{g,h}$ on $\mathcal{R}_p^c$ by

$$(x, y)_{g,h} = \sum_{i=1}^{c} x_i y_i^{r^g q^h}$$

where $x, y \in \mathcal{R}_p^c$ with $x = x_1 \cdots x_c$ and $y = y_1 \cdots y_c$.

If $x \in F_q^n$, let $x|_{\Omega_i}$ be the restriction of x to Ω_i. For $1 \le i \le c$, $x|_{\Omega_i}$ can be viewed as an element $x_{i,0} + x_{i,1}X + \cdots + x_{i,p-1}X^{p-1}$ of $\mathcal{R}_p$ where $x\sigma|_{\Omega_i} = x|_{\Omega_i}X$. So $\mathcal{C}$ can be viewed as an F_q-subspace of $\mathcal{R}_p^c \times F_q^f$. Let $\mathcal{C}(\sigma) = \{x \in \mathcal{C} \mid x\sigma = x\}$, and for $1 \le i \le t$, let $\mathcal{E}_i(\sigma) = \{x \in \mathcal{C} \mid x|_{\Omega_j} \in \mathcal{I}_i \text{ for } 1 \le j \le c \text{ and } x|_{\Omega_j} = 0 \text{ for } c+1 \le j \le c+f\}$. It can be shown [15] that $\mathcal{C}$ can be 'decomposed' as follows: $\mathcal{C} = \mathcal{C}(\sigma) \oplus \mathcal{E}_1(\sigma) \oplus \cdots \oplus \mathcal{E}_t(\sigma)$.

As we may essentially view each $\mathcal{E}_i(\sigma)$ as a code over the field $\mathcal{I}_i$, let $\mathcal{E}_i(\sigma)^*$ be the code $\mathcal{E}_i(\sigma)$ punctured on the fixed points $\Omega_{c+1} \cup \cdots \cup \Omega_{c+f}$ whose codewords are c-tuples with components in $\mathcal{I}_i$. Each $\mathcal{E}_i(\sigma)^*$ is linear over $\mathcal{I}_i$. The codewords of $\mathcal{C}(\sigma)$ are constant on cycles and so we can define the code $\mathcal{C}(\sigma)^*$ to be the code over F_q of length $c + f$ obtained from $\mathcal{C}(\sigma)$ by deleting all but one coordinate from each cycle.

A general result found in [16] produces restrictions on the 'components' of the decomposition of $\mathcal{C}$ in the three cases we are looking at. Knowing these restrictions assists in enumerating codes with automorphism σ.

In all cases $\mathcal{C}(\sigma)$ is a self-orthogonal code of dimension $\frac{c+f}{2}$. When $q = 2$ or $q = 4$, $\mathcal{C}(\sigma)^*$ is a self-dual $[c + f, \frac{c+f}{2}]$ code; if $\mathcal{C}$ is Type II and $p \equiv 1 \pmod 4$, then $\mathcal{C}(\sigma)^*$ is also Type II. When $\mathcal{C}$ is Type III and $p \equiv 1 \pmod 3$, then $\mathcal{C}(\sigma)^*$ is Type III as well.

The codes $\mathcal{E}_i(\sigma)^*$ of length c over $\mathcal{I}_i$ satisfy the following, depending on q and p. Two possibilities occur when q is 2 or 3. If $q^h \equiv -1 \pmod p$, then $\mathcal{I}_i \simeq F_{q^{2h}}$ and $\mathcal{E}_i(\sigma)^*$ is a $[c, \frac{c}{2}]$ code that is self-dual under the Hermitian type inner product $(\cdot,\cdot)_{0,h}$. If $q^h \not\equiv -1$

(mod p) for any h, then t is even and the map $\tau_{1,-1}$, which sends X to X^{-1}, interchanges $\mathcal{I}_1, \ldots, \mathcal{I}_t$ in pairs. Each field $\mathcal{I}_i$ is F_{q^b} where $b = (p-1)/t$. If $\mathcal{I}_i\tau_{1,-1} = \mathcal{I}_{i_\tau}$, then $\mathcal{E}_{i_\tau}(\sigma)^*$ and $\mathcal{E}_i(\sigma)^*\tau_{1,-1}$ are duals of each other under the ordinary inner product. When $q = 4$, three possibilities occur. Let a be the order of 2 modulo p. First, suppose that $2 \mid a$ but $4 \nmid a$. Then $\mathcal{I}_i \simeq F_{2^a}$ and $\mathcal{E}_i(\sigma)^*$ is a $[c, \frac{c}{2}]$ code that is self-dual under the Hermitian type inner product $(\cdot,\cdot)_{1,(a-2)/4}$. Next, suppose that a is odd. Then t is even, each field $\mathcal{I}_i$ is F_{4^a}, and the map $\tau_{1,-1}$, which again sends X to X^{-1}, interchanges $\mathcal{I}_1, \ldots, \mathcal{I}_t$ in pairs. If $\mathcal{I}_i\tau_{1,-1} = \mathcal{I}_{i_\tau}$, then $\mathcal{E}_{i_\tau}(\sigma)^*$ and $\mathcal{E}_i(\sigma)^*\tau_{1,-1}$ are duals of each other under the Hermitian type inner product $(\cdot,\cdot)_{1,(a-1)/2}$. Finally, suppose $4 \mid a$. Then t is even, each field $\mathcal{I}_i$ is F_{2^a}, and the map $\tau_{2,1}$, which is the conjugation map, interchanges $\mathcal{I}_1, \ldots, \mathcal{I}_t$ in pairs. If $\mathcal{I}_i\tau_{2,1} = \mathcal{I}_{i_\tau}$, then $\mathcal{E}_{i_\tau}(\sigma)^*$ and $\mathcal{E}_i(\sigma)^*\tau_{2,1}$ are duals of each other under the Hermitian type inner product $(\cdot,\cdot)_{0,a/4}$.

Under very reasonable circumstances, this code decomposition is preserved under equivalence [16]. Hence checking equivalence can be restricted to maps of a very specific form, making testing for equivalence relatively easy. In the case $q = 4$, a similar theory has been developed in [17] to examine codes which have an automorphism a power of 3.

n	d	order	number	reference
34	8	17	1	[13]
36	8	5	4	[45]
36	8	7	3	[37]
36	8	17	1	[13, 36]
38	8	5	0	[45]
38	8	7	7	[38]
38	8	19	1	[11]
40	8	5	37	[2]
40	8	7	2	[2]

n	d	order	number	reference
40	8	19	0	[11]
46	10	23	1	[11]
48	10	23	0	[11]
50	10	7	4	[21]
52	10	7	499	[22]
58	10	29	11	[12]
60	12	29	3	[13]
62	12	31	0	[11]
64	12	31	0	[11]

Table 5: Inequivalent optimal Type I codes classified by automorphism.

n	d	order	number	reference
40	8	4 5-cycles	1	[42]
40	8	7	5	[42]
40	8	19	3	[42]
48	12	odd	1	[14]
56	12	13	16	[43]
64	12	31	38	[44]
72	16	prime > 7	0	[4, 23, 31, 35]

Table 6: Inequivalent extremal Type II codes classified by automorphism.

The Type I and Type II codes that have been classified by automorphism (and known to the author) are presented in Tables 5 and 6. Note that codes of the same length but having different order automorphisms in these tables may be equivalent. There have also been efforts to completely classify extremal Type I or II codes which have circulant or double circulant generator matrices [11, 12, 13]. These include codes with automorphisms of order $\frac{n}{2}$ or $\frac{n}{2} - 1$ which may or may not be prime. Those that are prime are reflected in the tables. Table 6 reminds of the following problem many have looked at.

Research Problem 2.2 *Is there an extremal Type II code of length 72?*

n	d	order	number
28	9	7	14
28	9	13	5
28	9	5, 11, ≥ 17	0
32	9	5	239
32	9	7	16
32	9	≥ 11	0

n	d	order	number
36	12	17	1
36	12	$\geq 5, \neq 17$	0
40	12	13	4
40	12	19	11
40	12	7, 11, 17, ≥ 23	0

Table 7: Inequivalent extremal Type III codes classified by automorphism.

n	d	order	number
22	8	≥ 13	0
22	8	power of 3	38
22	8	5	9
22	8	7	2
22	8	11	3

n	d	order	number
26	10	power of 3, ≥ 5	0
28	10	power of 3	1
28	10	7	3
28	10	5, ≥ 11	0

Table 8: Inequivalent extremal Type IV codes classified by automorphism.

The Type III and IV codes that have been classified completely by automorphism are listed in Tables 7 (see [19]) and 8 (see [16, 18]). These tables suggest the following open problems.

Research Problem 2.3 *The Type III code of length* 36 *is a Pless symmetry code. Is it the unique extremal Type III code of that length?*

Research Problem 2.4 *Does there exist an extremal Type IV code of length* 26*?*

The study of self-dual linear $\mathbb{Z}_4$-codes, under the ordinary inner product moulo 4, has been taken up relatively recently. The self-dual codes of length up to 9 have been classified in [8]. (Unlike self-dual codes over F_q, self-dual codes over $\mathbb{Z}_4$ can have odd length.) The so called Type II $\mathbb{Z}_4$-codes of length 16 have been found in [10] using a mass formula and complete enumeration. The theory of 'decomposing' self-dual codes over F_q can be extended to self-dual $\mathbb{Z}_4$-linear codes, with some modifications. Assume the $\mathbb{Z}_4$ self-dual code $\mathcal{C}$ has permutation automorphism σ of prime order $p \neq 2$. The ring corresponding to $\mathcal{R}_p$ is $\mathbb{Z}_4[X]/(X^p-1)$. The polynomial $X^p-1 = M_0(X)M_1(X)\cdots M_t(X)$ where $M_i(X)$ are basic irreducibles with $M_0(X) = 1 - X$. The ideals corresponding to $\mathcal{I}_i$ are no longer fields but Galois ring extensions of $\mathbb{Z}_4$. The code $\mathcal{C}$ still decomposes as over F_q. Again there is a ring isomorphism τ that fixes $\mathcal{I}_0$ and permutes the remaining I_i. This map permutes the components $\mathcal{E}_i(\sigma)^*$ so that they are either self-dual, under a certain inner product, over a Galois ring or duals of each other when τ is applied. Under very reasonable circumstances, equivalence is also relatively easy to check. The author is currently examining the Type II $\mathbb{Z}_4$-codes of length 24 using this theory.

References

[1] E. F. Assmus, Jr., H. F. Mattson, Jr., and R. J. Turyn, "Research to develop the algebraic theory of codes," Report AFCRL-67-0365, Air Force Cambridge Res. Labs., Bedford, MA, June 1967.

[2] S. Buyuklieva and V. Y. Yorgov, "Singly-even self-dual codes of length 40," *Designs, Codes and Crypt.*, to appear.

[3] J. H. Conway and V. Pless, "On the enumeration of self-dual codes," *J. Comb. Theory* **A 28** (1980), 26–53.

[4] J. H. Conway and V. Pless, "On primes dividing the group order of a doubly-even (72,36,16) code and the group order of a quaternary (24,12,10) code," *Discrete Math.* **38** (1982), 143–156.

[5] J. H. Conway, V. S. Pless, and N. J. A. Sloane, "Self-dual codes over GF(3) and GF(4) of length not exceeding 16," *IEEE Trans. Inform. Theory* **IT-25** (1979), 312–322.

[6] J. H. Conway, V. S. Pless, and N. J. A. Sloane, "The binary self-dual codes of length up to 32: a revised enumeration," *J. Comb. Theory* **A 60** (1992), 183–195.

[7] J. H. Conway and N. J. A. Sloane, "A new upper bound on the minimal distance of self-dual codes," *IEEE Trans. Information Theory* **36** (1990), 1319-1333.

[8] J. H. Conway and N. J. A. Sloane, "Self-dual codes over the integers modulo 4," *J. Comb. Theory* **A 62** (1993), 30–45.

[9] J. H. Conway and N. J. A. Sloane, *Sphere Packings, Lattices, and Groups*, 2nd ed., New York: Springer-Verlag, 1993.

[10] J. Fields, J. S. Leon, and V. Pless, "All $\mathbb{Z}_4$ codes of Type II and Length 16 are known," preprint.

[11] T. A. Gulliver and M. Harada, "Weight enumerators of double circulant codes and new extremal self-dual codes," *Designs, Codes and Crypt.*, to appear.

[12] T. A. Gulliver and M. Harada, "Classification of extremal double circulant self-dual codes of lengths 64 to 72," preprint.

[13] M. Harada, T. A. Gulliver, and H. Kaneta, "Classification of extremal double circulant self-dual codes of length up to 62," preprint.

[14] W. C. Huffman, "Automorphisms of codes with applications to extremal doubly even codes of length 48," *IEEE Trans. Inform. Theory* **IT-28** (1982), 511–521.

[15] W. C. Huffman, "On the [24,12,10] quaternary code and binary codes with an automorphism having two cycles," *IEEE Trans. Inform. Theory* **IT-34** (1988), 486–493.

[16] W. C. Huffman, "On extremal self-dual quaternary codes of lengths 18 to 28, I," *IEEE Trans. Inform. Theory* **IT-36** (1990), 651–660.

[17] W. C. Huffman, "On 3-elements in monomial automorphism groups of quaternary codes," *IEEE Trans. Inform. Theory* **IT-36** (1990), 660–664.

[18] W. C. Huffman, "On extremal self-dual quaternary codes of lengths 18 to 28, II," *IEEE Trans. Inform. Theory* **IT-37** (1991), 1206–1216.

[19] W. C. Huffman, "On extremal self-dual ternary codes of lengths 28 to 40," *IEEE Trans. Inform. Theory* **IT-38** (1992), 1395–1400.

[20] W. C. Huffman, "Characterization of Quaternary Extremal Codes of Lengths 18 and 20," *IEEE Trans. Inform. Theory*, to appear.

[21] W. C. Huffman and V. D. Tonchev, "The existence of extremal [50, 25, 10] codes and quasi-symmetric 2-(49,9,6) designs," *Designs, Codes and Crypt.* **6** (1995), 97–106.

[22] W. C. Huffman and V. D. Tonchev, unpublished.

[23] W. C. Huffman and V. Y. Yorgov, "A [72, 36, 16] doubly even code does not have an automorphism of order 11," *IEEE Trans. Inform. Theory* **IT-33** (1987), 749–752.

[24] C. W. H. Lam and V. Pless, "There is no (24,12,10) self-dual quaternary code," *IEEE Trans. Inform. Theory* **IT-36** (1990), 1153–1156.

[25] J. S. Leon, V. Pless, and N. J. A. Sloane, "On ternary self-dual codes of length 24," *IEEE Trans. Inform. Theory* **IT-27** (1981), 176–180.

[26] F. J. MacWilliams, A. M. Odlyzko, N. J. A. Sloane, and H. N. Ward, "Self-dual codes over GF(4)," *J. Combin. Theory A* **25** (1978), 288–318.

[27] C. L. Mallows and N. J. A. Sloane, "An upper bound for self-dual codes," *Inform. and Control* **22** (1973), 188–200.

[28] C. L. Mallows, V. Pless, N. J. A. Sloane, "Self-dual codes over GF(3)," *SIAM J. Appl. Math.* **31** (1976), 649–666.

[29] V. Pless, "On the uniqueness of the Golay codes," J. Comb. Theory **5** (1968), 215–228.

[30] V. Pless, "A classification of self-orthogonal codes over GF(2)," *Discrete Math.* **3** (1972), 209–246.

[31] V. Pless, "23 does not divide the order of the group of a (72,36,16) doubly even code," *IEEE Trans. Inform. Theory* **IT-28** (1982), 113–117.

[32] V. Pless, "The children of the (32,16) doubly even codes," *IEEE Trans. Inform. Theory* **IT-28** (1982), 738–746.

[33] V. Pless and N. J. A. Sloane, "On the classification and enumeration of self-dual codes," *J. Comb. Theory* **18** (1975), 313–335.

[34] V. Pless, N. J. A. Sloane, and H. N. Ward, "Ternary codes of minimum weight 6 and the classification of the self-dual codes of length 20," *IEEE Trans. Inform. Theory* **IT-26** (1980), 305–316.

[35] V. Pless and J. G. Thompson, "17 does not divide the order of a group of a (72,36,16) code," *IEEE Trans. Inform. Theory* **IT-28** (1982), 537–541.

[36] R. P. Ruseva, "Uniqueness of the [36, 18, 8] double circulant code," in *Proc. of the International Workshop on Optimal Codes and Related Topics*, May 26–June 1, 1995, Sozopol, Bulgaria, 126–129.

[37] R. P. Ruseva, "New extremal self-dual codes of length 36," in *Proc. of the* 25th *Spring Conf. of the UBM*, 1996, 150–153.

[38] R. P. Ruseva, "On the extremal self-dual binary codes of length 38 with an automorphism of order 7," preprint.

[39] S. L. Snover, "The uniqueness of the Nordstrom–Robinson and the Golay binary codes," Ph.D. Thesis, Department of Mathematics, Michigan State Univ., 1973.

[40] H. N. Ward, "A restriction on the weight enumerator of self-dual codes," *J. Combin. Theory* **21** (1976), 253–255.

[41] H. N. Ward, "Divisible codes," Arch. Math. (Basel) **36** (1981), 485–494.

[42] V. Y. Yorgov, "Binary self-dual codes with automorphisms of odd order," *Prob. Inform. Trans.* **19** (1983), 11–24.

[43] V. Y. Yorgov, "A method for constructing inequivalent self-dual codes with applications to length 56," *IEEE Trans. Inform. Theory* **IT-33** (1987), 77–82.

[44] V. Y. Yorgov, "Doubly-even codes of length 64," *Prob. Inform. Trans.* **22** (1986), 35–42.

[45] V. Yorgov and N. Yankov, "On the extremal binary codes of lengths 36 and 38 with an automorphism of order 5," *Proc. of the* 5th *International Workshop on Algebraic and Combinatorial Coding Theory* June 1–7, 1996, Sozopol, Bulgaria, 307–312.

Techniques Used in the Classification of Self-Dual $\mathbb{Z}_4$ Codes

JOE FIELDS AND VERA PLESS

Mathematics Department, University of Illinois at Chicago
Chicago, Illinois 60607

fields@math.uic.edu, VPless@uic.edu

1. ABSTRACT

Recently [11], the self-dual $\mathbb{Z}_4$ codes of Type II and length 16 were completely classified. We discuss the methods used in this classification. We also present some of the codes with interesting properties that we found.

2. INTRODUCTION

Interest in $\mathbb{Z}_4$ codes has increased since it was shown [6] that "good" non-linear binary codes could be handled as Gray images of $\mathbb{Z}_4$ codes.

An interesting subclass of $\mathbb{Z}_4$ codes is the class of self-dual codes which exist at any length (in contrast to binary self-dual codes which can only exist at even lengths). There are classifications of all binary, ternary and quaternary self-dual codes of modest lengths. Such classifications are possible because there are "mass formulas" for all of these classes of codes. (A mass formula gives the total number of codes having a specific property.) Conway and Sloane [3] classified all self-dual $\mathbb{Z}_4$ codes of length up to 9 without a mass formula. Recently Gaborit [5] has demonstrated a mass formula for both the self-dual and the Type II $\mathbb{Z}_4$ codes. Using this information and a new code automorphism group program [7, 8], we have classified all Type II $\mathbb{Z}_4$ codes of length 16 [11]. Generator matrices for all of these codes are available on the world wide web at `http://www.math.uic.edu/~pless`

We found that there are 133 inequivalent Type II codes of length 16. As Type II codes can only exist at lengths divisible by 8, this is the next classification possible after the work of Conway and Sloane at length 8 – where there were just 4 Type II codes.

In our classification we were often in the situation where there were too many codes of a specific form to be distinguished by the automorphism group program. In fact there were up to 2^{21} codes of some types. We were able to reduce these numbers significantly so that at most a few hundred were run at a time with the code automorphism group program.

We discuss the methods used to make this reduction. Some involved finding representatives of certain $n \times n$ matrices under the action of S_n. Others involved finding elements in the dual of a certain subcode – these elements could be added to some of a

code's generators producing the canonical form of the generator matrix for an equivalent code. We will discuss these methods more fully in Section 3. We will also discuss the linearity of the Gray images of the 5 inequivalent codes attaining the highest minimum Lee weight, 8.

3. Preliminaries

A $\mathbb{Z}_4$ code is a module over the ring $\mathbb{Z}/(4\mathbb{Z})$.

Two $\mathbb{Z}_4$ codes are *equivalent* if one can be obtained from the other by permuting coordinates and making sign changes in a subset of the coordinates.

For $\mathbb{Z}_4$ codes, the usual notion of dimension does not hold. Every $\mathbb{Z}_4$ code C is equivalent (indeed, equivalent via permutations only) to a code having a generator matrix of the form

$$\begin{bmatrix} I_{j\times j} & A & B \\ 0 & 2I_{k\times k} & C \end{bmatrix}$$

and C is said to have *type* 4^j2^k [3]. Furthermore, the number of vectors in such a code is 4^j2^k.

Orthogonality is defined as usual, and a code C is *self-dual* if $C = C^{\perp}$.

The *Lee weight* of a $\mathbb{Z}_4$ vector is an integer that counts 1 and 3 components as one, and 2 components as two. In computing the *Euclidean weight* of a vector, a 2 component counts as four. A $\mathbb{Z}_4$ code is of *Type II* if the Euclidean weight of any vector in it is divisible by 8 and it contains a vector equivalent to the all-ones vector.

The automorphism group of a length n code over $\mathbb{Z}_4$ is the group of all coordinate permutations and sign changes that fix C set-wise. These groups have natural permutation representations on $2n$ points. There is a normal subgroup consisting of sign changes only, and the quotient group acts naturally on n points.

In [11] we showed that any Type II code C of length n has a generator matrix of the form

$$G = \begin{bmatrix} A & \tilde{I}_k + 2B \\ & 2C \end{bmatrix} \tag{1}$$

where A,B and C are binary matrices, $\tilde{I}_k = \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 0 & & & \\ \vdots & & I_{k-1} & \\ 0 & & & \end{bmatrix}$, and all of the following hold:

- $\begin{bmatrix} A & \tilde{I}_k \end{bmatrix}$ is a generator matrix for a doubly-even binary code C_1 of length n.
- $\begin{bmatrix} A & \tilde{I}_k \\ & C \end{bmatrix}$ generates $C_2 = C_1^{\perp}$.

- The matrix B is chosen to make C self-orthogonal over $\mathbb{Z}_4$ and so that every vector has Euclidean weight divisible by 8.

The code C has type $4^k 2^{n-2k}$.

We labeled a Type II code by describing the weight 4 vectors in C_1 and by giving its type. We recall that every self-orthogonal binary code generated by weight 4 vectors is a direct sum of d_{2n} ($n \geq 2$), e_7 and e_8 [2]. If C_1 does not contain any weight 4 vectors, it receives the label f_t, where t is its dimension.

For each doubly-even code C_1 of length 16 and dimension k, there are as many $\mathbb{Z}_4$ Type II codes with C_1's label as there are ways to fill in the matrix $\tilde{I}_k + 2B$ in such a way as to satisfy the Type II constraints.

It is not difficult to find all the inequivalent doubly-even binary codes of length 16 and dimension k where $1 \leq k \leq 8$. For many of these doubly-even codes C_1, it is difficult to reduce the number of inequivalent $\mathbb{Z}_4$ codes C (with this C_1 and a given type) so that they can be handled by the group automorphism program. We describe how to do this reduction in the next section.

We use the canonical generator matrix (1) to see all of the choices and perform the reductions.

4. Methods

We call a doubly-even binary code which contains the all-ones vector *admissible*. Up to equivalence there are four admissible codes of length 16 and dimension 3. We label these codes $3-d_4$, $3-2d_4$, $3-d_6$ and $3-f_3$. These codes have (respectively) a single vector of weight four, two disjoint vectors of weight four, three mutually intersecting weight four vectors, and no weight four vectors at all.

From each of these admissible codes we obtain $\mathbb{Z}_4$ codes with the same basic label by extending the basis of the code to one for its dual (the vectors added in this extension are multiplied by 2) and filling in the off-diagonal positions (except those in the first row) in $\tilde{I}_2 + 2B$ in all possible ways (subject to the Type II constraints). In the proof of the mass formula, the positions on the diagonal are also taken to be free – since we are concerned with inequivalent codes, we fix these positions (a sign change on the column containing a 3 on the diagonal of $\tilde{I}_k$ would give us an equivalent code with a 1 there). Hence each of these admissible codes gives two $\mathbb{Z}_4$ codes.

For example, the admissible code $3-d_6$ has generator matrix

$$G_{3-d_6} = \left[\begin{array}{cccccccccccccc|ccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right].$$

From this code we get the following two $\mathbb{Z}_4$ codes of Type II:

$$G_{\mathbf{3-d_6a}} = \left[\begin{array}{cccccccccccc|ccc}
1&1&1&1&1&1&1&1&1&1&1&1&1&3&3\\
1&1&1&0&0&0&0&0&0&0&0&0&0&2&1&0\\
1&1&0&1&0&0&0&0&0&0&0&0&0&0&2&1\\
\hline
2&2&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
2&0&2&2&2&0&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&2&2&0&0&0&0&0&0&0&0&0&0\\
0&0&0&0&2&0&2&0&0&0&0&0&0&0&0&0\\
0&0&0&0&2&0&0&2&0&0&0&0&0&0&0&0\\
0&0&0&0&2&0&0&0&2&0&0&0&0&0&0&0\\
0&0&0&0&2&0&0&0&0&2&0&0&0&0&0&0\\
0&0&0&0&2&0&0&0&0&0&2&0&0&0&0&0\\
0&0&0&0&2&0&0&0&0&0&0&2&0&0&0&0\\
0&0&0&0&2&0&0&0&0&0&0&0&2&0&0&0
\end{array}\right]$$

and

$$G_{\mathbf{3-d_6b}} = \left[\begin{array}{cccccccccccc|ccc}
1&1&1&1&1&1&1&1&1&1&1&1&1&1&3&3\\
1&1&1&0&0&0&0&0&0&0&0&0&0&0&1&2\\
1&1&0&1&0&0&0&0&0&0&0&0&0&2&0&1\\
\hline
2&2&0&0&0&0&0&0&0&0&0&0&0&0&0&0\\
&&&&&&\vdots&&&&&&&&\vdots&\\
0&0&0&0&2&0&0&0&0&0&0&0&2&0&0&0
\end{array}\right].$$

A note on notation: admissible codes are binary, they are distinguished from their $\mathbb{Z}_4$ descendents by using boldface type for the labels of the latter.

Notice two things, these generator matrices are identical except for their upper-right 3×3 blocks, and furthermore they span equivalent codes. We can see this equivalence in two different ways.

First, note that the vector $\left[\begin{array}{cccccccccccccc|ccc} 2&0&0&0&0&0&0&0&0&0&0&0&0&0&2&2&2 \end{array}\right]$ is in both codes (this may not be immediately obvious, but one only needs to verify that it is orthogonal to the first three rows of each generator matrix). If we add this vector to each of the first three rows of (say) $G_{\mathbf{3-d_6a}}$, then make the appropriate sign changes (in columns 1, 14, 15 and 16) we have converted one generator matrix to the other.

The second way of seeing the equivalence of these codes is to perform row and column permutations (interchange columns 15 and 16, then interchange rows 2 and 3, finally interchange columns 3 and 4).

In the remainder of this paper we will not give full generator matrices for $\mathbb{Z}_4$ codes, we will only give the first k rows of a generator matrix (where k is the exponent of 4 in the type of the code). We call the $k \times k$ block at the right-hand side of these partial generator matrices the Gaborit square, or more succinctly, the G-square. The remainder (i.e. the

part that is not shown) of the generator matrices we call the 2-part – it is completely determined by the binary code C_1. Thus the generator matrices of codes that are derived from the same binary code will only differ in their G-squares.

The two codes derived from $3 - d_4$ have the following (partial) generator matrices:

$$\left[\begin{array}{cccccccccccccc|ccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 3 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 1 \end{array}\right]$$

and

$$\left[\begin{array}{cccccccccccccc|ccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 3 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 2 \\ 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 2 & 0 & 1 \end{array}\right]$$

The vector $\left[\begin{array}{cccccccccccccc|ccc} 0 & 2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & 2 \end{array}\right]$ is orthogonal to the displayed portion of each of these generator matrices, and so is in both codes. The addition of this vector to all three rows of either matrix will transform it into a matrix equivalent to the other (sign changes must be made in columns 2, 14, 15 and 16).

The first case (that we encountered) where inequivalent $\mathbb{Z}_4$ codes are derived from the same admissible code is $3 - 2d_4$, these $\mathbb{Z}_4$ codes are thus given the labels $\mathbf{3 - 2d_4a}$ and $\mathbf{3 - 2d_4b}$. These codes have different symmetrized weight enumerators.

There are 8 admissible codes of dimension 4 and length 16 having labels: $4 - 2d_4$, $4 - 4d_4$, $4 - d_4 + d_6$, $4 - d_4$, $4 - d_6$, $4 - d_8$, $4 - e_7$ and $4 - f_4$. We can construct 2^3 $\mathbb{Z}_4$ codes having each of these labels. Using the techniques described above we can show that up to equivalence there is exactly one $\mathbb{Z}_4$ code derived from each of the codes: $4 - d_4$, $4 - d_6$, $4 - e_7$ and $4 - f_4$. From the code $4 - 4d_4$, we get three $\mathbb{Z}_4$ codes, and from the remaining admissible codes ($4 - 2d_4$, $4 - d_4 + d_6$, $4 - d_8$) we get two.

Let us consider the codes derived from $4 - d_8$. We take a generator matrix for $4 - d_8$ to be

$$G_{4-d_8} = \left[\begin{array}{cccccccccccccc|cccc} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{array}\right],$$

and note the symmetries of this matrix: the first row is distinct, but (up to equivalence) we can freely permute the last three rows. Since the inner products among these last three rows are all two, the relevant entries in the G-square must be skew-symmetric. Clearly S_3 acts on the last three rows and columns of the G-square, and so, we need to find representatives for the orbits of this action. We can find these representatives computationally – a brute force approach is to make a list of all possible G-squares, compute the orbit of the first element in the list, place this first element in a new list of

orbit representatives, remove all elements of this orbit from the list and repeat until the list is exhausted.

A slightly less brutal approach is to take advantage of a one-to-one correspondence between G-squares and digraphs.

Essentially, to every binary admissible code, we associate a set of graphs the points of which correspond to basis elements for the code. That is, the points correspond to rows of a generator matrix in canonical form (aside from the first row, where we always have the all-ones vector). Edges come in two classes: if the inner-product of the vectors corresponding to the endpoints is singly-even the edge must be directed, and if the inner-product is doubly-even the edge must be undirected. Different graphs that correspond to a given admissible code have the same directed/undirected 'signature'. Certain choices are made to determine a particular graph: when an edge must be directed, we choose the direction, and when an edge must be undirected we choose whether to put it in or not. These graphs look like multi-partite graphs in which the 'parts' consist of complete directed graphs, and some subset of the possible (undirected) edges between parts.

The G-square is a matrix of the form $\tilde{I}_k + 2B$. Consider the sub-matrix B' which consists of the last $k-1$ rows and columns of B. B' is the adjacency matrix of this directed graph. A 1 in the ith row and jth column of B' indicates that there is an edge going from vertex j to vertex i.

How many essentially distinct complete digraphs are there on three points?

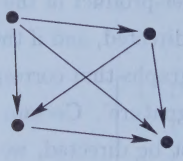
FIGURE 1. There are just two complete digraphs on 3 points.

This sort of situation occurs many times and so we have found the G-squares which correspond to the 4 complete digraphs on 4 points, the 12 complete digraphs on 5 points and the 56 complete digraphs on 6 points. For instance, the code $5 - d_{10}$ has (a priori) 2^6 $\mathbb{Z}_4$ descendents, but we can reduce this immediately to the four representatives of S_4 acting on skew-symmetric matrices by row and column permutations. Interestingly, it turns out that there are only two inequivalent $\mathbb{Z}_4$ codes descendant from $5 - d_{10}$, but our purpose here was only to reduce the number of codes that need to be distinguished by the automorphism group program.

It would appear that the component code e_7 would, like d_8, have two $\mathbb{Z}_4$ descendents. This is not so – and provides an illustration of our other main technique. First, let us consider d_8 again. Since there is a column of 1's in the generator matrix for d_8, there is a special element in $2C_2$ (Recall that C_2 is $C_1^{\perp}$), namely

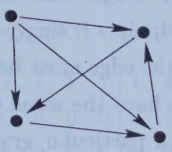
$$s = [2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 2, 2, 2].$$

If we add s to every row of a generator matrix for some $\mathbb{Z}_4$ descendant, and then make a sign change in the first column, we have a generator matrix (back in canonical form) for an equivalent code. In the 'graphs' context, s gives us the freedom to (simultaneously) reverse all the arrows. Such an element (i.e. one that gives us the "reverse-all-arrows" action) exists for all d_{2n} codes. While this action doesn't collapse any orbits for d_8, it does for d_{10} (and larger codes, see Figure 2) – where certain digraphs become equivalent, as one is the result of reversing all of the arrows in the other.



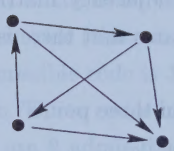
1 1 1 1 1
* 1 0 0 0
* 2 1 0 0
* 2 2 1 0
* 2 2 2 1

Source-Sink



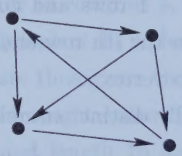
1 1 1 1 1
* 1 0 0 0
* 2 1 0 2
* 2 2 1 0
* 2 0 2 1

Source only



1 1 1 1 1
* 1 0 2 0
* 2 1 0 0
* 0 2 1 0
* 2 2 2 1

Sink only



1 1 1 1 1
* 1 0 0 2
* 2 1 0 0
* 2 2 1 0
* 0 2 2 1

Balanced

FIGURE 2. The graphs labeled "source only" and "sink only" are equivalent under the 'reverse-all-arrows' action.

Back to e_7, here there are other elements of $2C_2$ that can be used to "reverse any arrow". For instance the vector

$$s = [0, 2, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 2, 0, 2, 2]$$

can be added to the 1st, 3rd and 4th rows of a generator matrix for a descendant code:

$$G_{4-e_7*} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 3 & 3 & 3 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 1 & * \\ 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 2 & 2 & * & 1 \end{bmatrix}$$

and after appropriate sign changes on columns, we have a generator matrix for an equivalent code (in canonical form) where a different choice for the starred positions has been made. There are vectors in $2C_2$ that give us such freedom for all the positions in the G-square of an e_7 descendant code – thus there is only one descendant (up to equivalence).

Another situation that occurs frequently can be illustrated with the admissible code $6 - d_6 + e_7$. A generator matrix for this code is

$$G_{6-d_6+e_7} = \left[\begin{array}{cccc|cccccc|c|cc|ccc} 1&1&1&1&1&1&1&1&1&1&1&1&1&1&1&1 \\ \hline 1&1&1&0&0&0&0&0&0&0&0&1&0&0&0&0 \\ 1&1&0&1&0&0&0&0&0&0&0&0&1&0&0&0 \\ \hline 0&0&0&0&1&0&1&1&0&0&0&0&0&1&0&0 \\ 0&0&0&0&1&1&0&1&0&0&0&0&0&0&1&0 \\ 0&0&0&0&1&1&1&0&0&0&0&0&0&0&0&1 \end{array}\right],$$

which has been partitioned to show the block structure. From the analysis done previously, we would expect that there could only be one component subcode corresponding to the d_6 block (the number of distinct complete digraphs on two points). We have just seen that there is only one possibility for the 3×3 block that corresponds to the e_7 component subcode. But what about the 3×2 and 2×3 off-diagonal blocks? Clearly if these are zero, we have a direct sum, and the all-ones vector serves to 'glue' the two components together. What other possibilities are allowed?

Since (up to equivalence) we are free to permute the rows and columns of the G-square that correspond to a particular component code amongst themselves, we have an action of $S_2 \times S_3$ on the square.

Notice that vectors from different components have inner-product zero – so the entries in these off-diagonal blocks must be symmetric. In the context of our digraph analogy, this means that edges go in both directions between points that are in different blocks, thus we can think of these edges as being undirected.

We are free to permute the points that correspond to basis elements of a component subcode amongst themselves, but not between different component subcodes. The action of $S_2 \times S_3$ on the graph respects this partition.

It is a tedious exercise to determine the inequivalent graphs that have a certain directed/undirected signature, an example is given in Figure 3.

It should be pointed out that there are in fact only three $\mathbb{Z}_4$ descendents of $6-d_6+e_7$, so the correspondence with digraphs is not exact. That is, there are equivalent codes which correspond to different digraphs. Nevertheless, 13 (see Figure 3) is a significant improvement over the 1024 codes that would have had to have been checked for isomorphisms a priori.

5. Five codes with minimum Lee weight 8

The highest minimum Lee weight in a Type II code of length 16 is 8, and there are exactly 5 codes with this minimum weight.

We see that these codes have types $4^5 2^6$, $4^6 2^4$, $4^7 2^2$ and 4^8. The two codes of type 4^8 have identical symmetrized weight enumerators and their groups have the same orders

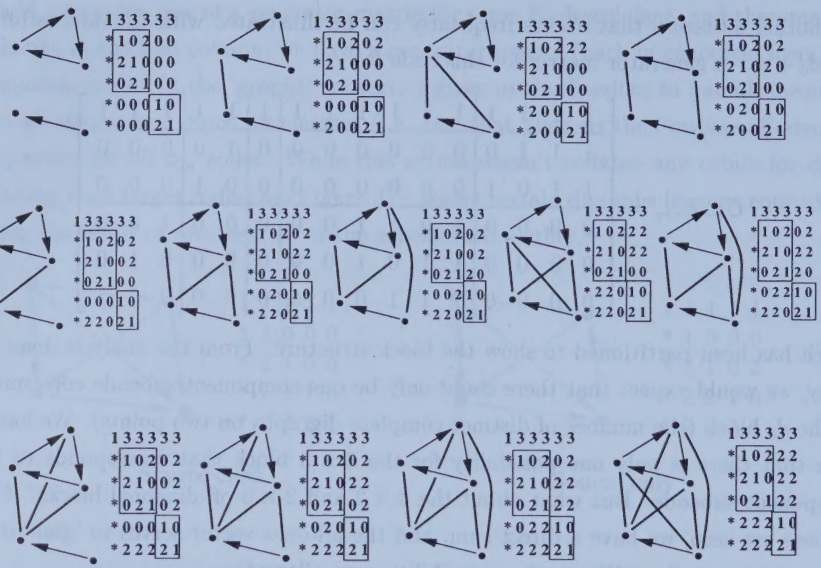


FIGURE 3. There are 13 distinct graphs having the directed/undirected signature of $6 - d_6 + e_7$.

FIGURE 4. Symmetrized weight enumerators for the five codes having minimum Lee weight 8. (Reproduced from [11])

label	swe coefficients										
	1 c^{16}	c^4 c^{12}	c^6 c^{10}	c^8	b^4c^3 b^4c^9	b^4c^5 b^4c^7	b^8 b^8c^8	b^8c^2 b^8c^6	b^8c^4	$b^{12}c$ $b^{12}c^3$	b^{16}
5_f5	1	140	448	870	0	0	480	13440	33600	0	2048
6_4d4c	1	76	192	486	512	1536	480	11904	30528	2048	1024
7_2d8g	1	44	64	294	768	2304	480	11136	28992	3072	512
8_d16g	1	28	0	198	896	2688	480	10752	28224	3584	256
8_2e8j	1	28	0	198	896	2688	480	10752	28224	3584	256

and orbit structure. But, two codes with different labels cannot be equivalent. The admissible code $5 - f$ is actually the unique $[16, 5, 8]$ code $R(1, 4)$. From the table above we see that the Gray image C of $5 - f5$ contains only vectors of weights divisible by 4.

We will need the following lemma:

Lemma *If C is a self-dual $\mathbb{Z}_4$ code whose Gray image $G(C)$ consists entirely of doubly-even vectors, then $G(C)$ is a linear binary code.*

Proof. It is known [6] that the Gray map is an isometry – if x and y are $\mathbb{Z}_4$ codewords so that the Lee weight of $x - y$ is d, then the Hamming weight of $G(x) - G(y)$ is also d.

Consider two vectors $x, y \in G(C)$, say $x = G(a)$ and $y = G(b)$ for $a, b \in C$. The distance between a and b (i.e. the Lee weight of $a - b$) is doubly-even since $G(a - b) \in$

$G(C)$. The distance between x and y (i.e. the Hamming weight of $x + y$) must also be doubly-even. Using the formula

$$|x + y| = |x| + |y| - 2(x \cap y)$$

for the weight of the sum of two vectors, we can deduce that $x \cap y$ is even, hence x and y are orthogonal. So $G(C)$ is a set of mutually orthogonal vectors, and so the linear code that $G(C)$ generates is self-orthogonal. But $G(C)$ is already as big as is possible for a self-orthogonal subspace of $\mathbb{F}_2^{2n}$.

□

Because the binary code $G(5 - f5)$ has only doubly-even vectors (see Figure 4), by the lemma it is linear. There are 5 inequivalent doubly-even $[32, 16, 8]$ codes [2]. It is not hard to see that $G(5 - f5)$ is $R(2, 5)$.

If we compute the number of singly-even and doubly-even vectors in the Gray images of the other 4 codes it is easy to determine that they do not satisfy the conditions in [10] for these codes to be linear. Hence the Gray images of the last four codes listed above are $(32, 2^{16}, 8)$ non-linear binary codes.

References

[1] A. Bonnecaze, P. Sole, Quaternary type II codes, to appear.

[2] J.H. Conway, V. Pless, On the Enumeration of Self-Dual Codes *J. Comb. Theory* **A 28** (1980), 26-53.

[3] J.H. Conway, N.J.A. Sloane, Self-Dual Codes over the Integers Modulo 4, *J. Comb. Theory* **A 62** (1993), 31-45.

[4] J. Fields, V. Pless, J. Leon, The Self-Dual $\mathbb{Z}_4$ Codes of Length Up To 15, preprint.

[5] P. Gaborit, Mass Formulas for Self-Dual Codes Over $\mathbb{Z}_4$ and $\mathbb{F} + u\mathbb{F}_q$ Rings, *IEEE Trans. Inform. Theory* **IT42 n. 4** (1996), 1222-1228.

[6] A.R. Hammons, Jr., P.V. Kumar, A.R. Calderbank, N.J.A. Sloane and P. Sole, The $\mathbb{Z}_4$ Linearity of Kerdock, Preperata, Goethals and Related Codes, *IEEE Trans. Inform. Theory* **IT40** (1994), 301-319.

[7] Jeffrey S. Leon, Permutation Group Algorithms Based on Partitions, I: Theory and Algorithms, *J. Symbolic Computation.*

[8] Jeffrey S. Leon, Partitions, Refinements, and Permutation Group Computations, *Proc. DIMACS Workshop on Groups and Computation*, to appear.

[9] V. Pless, A Classification of Self-Orthogonal Codes over GF(2), *Discrete Math.* **3** (1972), 209-246.

[10] V. Pless, Parents, Children, Neighbors and the Shadow, *Contemporary Math.* **168** (1994), 279-290.

[11] V. Pless, J. Leon, J. Fields, All $\mathbb{Z}_4$ Type II Codes of Length 16 are Known, to appear in JCT(A).

Lattices and codes over rings

Iwan M. Duursma & Joseph J. Boutros
duursma@research.att.com, boutros@com.enst.fr

We analyze examples of an infinite family of lattices that was defined by the first author in [5]. The paper [5] relates the lattices to $\mathbf{Z}4$-linear Preparata codes [6], to modular cyclic codes [4], and to low-correlation sequences [7]. Some of the lattices have relatively high fundamental gain with a lower kissing number than the densest lattices. They are suitable for digital transmission over AWGN channels. Other lattices have the same kissing number as $\mathbf{Z}^n$ but a higher gain. They can be used for digital transmission over Rayleigh fading channels [2].

The lattices are defined with a finite field F and a positive integer d. The size of F defines the dimension of the lattice. For a field F of size p^m, let the integer e be such that $p^{e-1} < d \leq p^e$. The lattice $L(p^m, d)$ is defined as the set of all $(n_\alpha)_{\alpha \in F}$, for $n_\alpha \in \mathbf{Z}$, such that

$$\prod_{\alpha \in F} (1 + \alpha T)^{n_\alpha} = 1 + a_d T^d + a_{d+1} T^{d+1} + \cdots. \tag{1}$$

$$\sum_{\alpha \in F} n_\alpha = 0 \in \mathbf{Z}/p^e\mathbf{Z}. \tag{2}$$

From the definitions (1,2), it follows that

$$p^e \mathbf{Z}^{p^m} \subset L(p^m, d) \subset \mathbf{Z}^{p^m}. \tag{3}$$

Furthermore, the lattice is invariant under the coordinate permutations

$$\alpha \mapsto a\,\alpha + b, \quad \text{for } a \in F - \{0\}, \ b \in F. \tag{4}$$

Tables 1 and 2 describe the lattices $L(2^m, 3)$. Each lattice has $p^e = 4$. With (3), the quotients $L(2^m, 3)/4\mathbf{Z}^{2^m}$ define a family of $\mathbf{Z}4$-linear codes of length 2^m. In [5], we show that these are the $\mathbf{Z}4$-linear Preparata codes [6]. And we obtain the tables from the complete weight enumerators of the Preparata codes [1].

Tables 3, 4 and 5 describe lattices $L(2^m, d)$ of fixed dimension $2^m = 8, 16$ and 32 respectively. Some of the listed lattices have interesting characteristics. $L(8, 2)$ and $L(8, 3)$ are equivalent to the Gosset lattice E_8. $L(16, 2)$ and $L(16, 6)$ (resp. $L(32, 4)$ and $L(32, 11)$) show a fundamental gain very close to Λ_{16} (resp. Λ_{32}) with a lower error coefficient. These dense lattices may exhibit a better performance than those of Barnes-Wall, especially at moderate signal-to-noise ratios [2]. An exceptional case is $L(32, 13)$ $(= L(32, 14))$. This lattice has the same kissing number as $\mathbf{Z}^{32}$ and a gain of 3.95dB. A finite constellation of $L(32, 13)$ can be labeled with a Gray mapping and combined to the rotated lattice $\mathbf{Z}_{32,16}$ [3] to yield good performance over a Rayleigh fading channel. The lattice $L(16, 7)$ can be applied in the same manner as $L(32, 13)$.

References

[1] A. Bonnecaze, I.M. Duursma: "Translates of $\mathbf{Z}_4$-linear codes," *IEEE Trans. on Information Theory*, To appear.

m	${d_E}^2$	$vol(L)$	gain (dB)
1	16	2^4	0.00
3	8	2^8	3.01
$5, 7, \ldots$	6	2^{2m+2}	7.78 $(m \to \infty)$
$2, 4, \ldots$	4	2^{2m+2}	6.02 $(m \to \infty)$

Table 1: Parameters for the Preparata lattices $L(2^m, 3)$

m	${d_E}^2$	$P \in B_{d_E}(0) \backslash 0$	#
1	16	$(\pm 4)^1$	4
3	8	$(1)^3(\pm 2)^1(-1)^1$	112
		$(1)^1(\pm 2)^1(-1)^3$	112
		$(1)^4(-1)^4$	14
		$(1)^8$	1
		$(-1)^8$	1
$5, 7, \ldots$	6	$(1)^5(-1)^1$	$2^m(2^m-1)(2^m-2)(2^m-8)/120$
		$(1)^1(-1)^5$	$2^m(2^m-1)(2^m-2)(2^m-8)/120$
		$(1)^3(-1)^3$	$2^m(2^m-1)(2^m-2)(2^m-8)/36$
$2, 4, \ldots$	4	$(1)^4$	$2^m(2^m-1)/12$
		$(-1)^4$	$2^m(2^m-1)/12$

Table 2: The first shell of the Preparata lattices $L(2^m, 3)$

[2] J. Boutros, E. Viterbo, C. Rastello, J. C. Belfiore: "Good lattice constellations for both Rayleigh fading and Gaussian channels," *IEEE Trans. on Information Theory*, vol. 42, no. 2, pp. 502–518, March 1996.

[3] J. Boutros, E. Viterbo: "Signal space diversity: a new power and bandwidth efficient diversity technique for the fading channel," *IEEE Trans. on Information Theory*, To appear.

[4] A.R. Calderbank, N.J.A Sloane: "Modular and p-adic cyclic codes," *Designs, Codes and Cryptography*, vol. 6, pp. 21–35, 1995.

[5] I.M. Duursma: "Preparata codes through lattices," *IEEE Trans. on Information Theory*, To appear.

[6] R. Hammons, P.V. Kumar, A.R. Calderbank, N.J.A Sloane, P. Solé: "The $\mathbf{Z}_4$-linearity of Kerdock, Preparata, Goethals, and related codes," *IEEE Trans. on Information Theory*, vol. 40, no. 2, pp. 301–319, March 1994.

[7] P.V. Kumar, T. Helleseth, A.R. Calderbank: "An upper bound for Weil exponential sums over Galois rings and applications," *IEEE Trans. on Information Theory*, vol. 41, no. 2, pp. 456–468, March 1995.

d	$d_E{}^2$	$vol(L)$	gain (dB)	τ
1	1	2^0	0.00	16
2	4	2^4	3.01	240
3	8	2^8	3.01	240
4	8	2^{12}	0.75	16

Table 3: Parameters for the lattices $L(2^3, d)$

d	$d_E{}^2$	$vol(L)$	gain (dB)	τ
1	1	2^0	0.00	32
2	4	2^5	4.13	2272
3	4	2^{10}	2.25	40
4	6	2^{14}	2.51	96
5	10	2^{19}	2.85	96
6	16	2^{21}	4.13	2272
7	16	2^{25}	2.63	32
8	16	2^{29}	1.12	32

Table 4: Parameters for the lattices $L(2^4, d)$

d	$d_E{}^2$	$vol(L)$	gain (dB)	τ
1	1	2^0	0.00	64
2	4	2^6	4.89	19904
3	6	2^{12}	5.52	31744
4	8	2^{17}	5.83	49600
5	10	2^{23}	5.67	19840
6	12	2^{28}	5.52	33728
7	12	2^{33}	4.58	992
8	16	2^{38}	4.89	1984
9, 10	24	2^{44}	5.52	9920
11	32	2^{49}	5.83	51648
12	32	2^{54}	4.89	19904
13, 14	32	2^{59}	3.95	64
15	32	2^{64}	3.01	64
16	32	2^{69}	2.06	64

Table 5: Parameters for the lattices $L(2^5, d)$

A New Efficient Method for Optimization of a Class of Nonlinear Dynamic Systems Without Lagrange Multipliers

Sunil K. Agrawal, Associate Professor
Nadeem Faiz, Graduate Student

Mechanical Systems Laboratory
Department of Mechanical Engineering
University of Delaware, Newark, DE 19716

Abstract

This paper deals with optimization of a class of nonlinear dynamic systems with n states and m control inputs commanded to move between two fixed states in a prescribed final time. Using conventional procedures with Lagrange multipliers, it is well known that the optimal trajectory is the solution of a two-point boundary value problem. The solution is obtained using multiple shooting methods which are computation intensive. The procedure also requires a good guess for Lagrange multipliers. Unfortunately, Lagrange multipliers can not be guessed since they are artificial variables.

In this paper, a new procedure for dynamic optimization of this problem is presented that does not use Lagrange multipliers. The procedure relies on the tools of feedback linearization where it is possible to transform classes of nonlinear dynamic systems into linear systems through nonlinear change of coordinates and control. The new form is such that the states and control can be written as higher derivatives of a subset of the states. Using this new form, it is possible to change the constrained dynamic optimization problem into an unconstrained dynamic optimization problem. The fundamentals of calculus of variations are then applied to this new cost functional and the necessary conditions for optimality are obtained. These necessary conditions are then efficiently solved using weighted residual methods. The computational efficiency of this procedure makes it attractive for real-time implementation.

1 Introduction

The dynamic systems considered in this paper have the form:

$$\dot{X} = f(X) + \sum_{i=1}^{m} g_i(X) u_i \qquad (1)$$

where X is a $(n \times 1)$ state vector, $f(X)$, $g_i(X)$ are smooth vector fields in $\mathcal{R}^n$, and u_i are scalar control inputs. It is assumed here that n is an integer multiple of m, i.e., $n = pm$. The above dynamic system is written more compactly as

$$\dot{X} = f(X) + G(X)U \qquad (2)$$

where $G(x)$ is a $n \times m$ matrix function obtained by stacking m columns of $g_i(X)$ and U is $m \times 1$ vector consiting of u_i.

The problem statement is to determine $U(t)$ that takes the system from an initial state X_0 at time t_0 to a final state X_f at t_f such that the path during t_0 and t_f minimizes the following cost functional:

$$J = \int_{t_0}^{t_f} \bar{F}(X, U) dt \qquad (3)$$

The problem described here is refered to as the fixed end time and fixed end state problem. Towards the end of the paper, extensions of the method proposed here are discussed for problems with constrained end conditions, or where n is not an integer multiple of m.

The conventional way of handling this problem ([4], [8], [2]) is to augment the state Eqs. (1) to the cost functional J using Lagrange multipliers. Using variational calculus, the necessary conditions for extremum are then written. These necessary conditions are $2n$ first-order differential equations and m algebraic equations involving the states, Lagrange multipliers, and controls. The boundary conditions are given state variables at t_0 and t_f. The two-point boundary-value problem (TPBVP) is solved iteratively using multiple shooting.

In multiple shooting, the time domain is partitioned into N intervals to result in $N + 1$ nodes. At each node, the unknowns are the states, Lagrange multipliers, and control inputs. A solution satisfies (i) the algebraic constraints at the nodes, (ii) the given end conditions, and (iii) continuity of state and Lagrange variables across the nodes, i.e., the result at a node $i + 1$ obtained by integrating the $2n$ differential equations with the guess of node i is the same as the guess of node $i + 1$. In essence, the two-point boundary value problem reduces to solution of $(2n + m)(N + 1)$ nonlinear equations in the same number of variables.

These nonlinear equations are solved using first-order gradient methods.

From this description of multiple-shooting procedure, the following comments can be made: (i) solution of the nonlinear equations is highly sensitive to the initial guess of the variables. Unfortunately, there is no logical way to guess the Lagrange multiplier variables within the domain. (ii) N is usually large, requiring solution of large systems of nonlinear equations. The solution also requires computation and inversion of Jacobians, which are $[(2n+m)(N+1)\times(2n+m)(N+1)]$ matrices, at every iteration step. (iii) initial-value problem for the $2n$ differential equations must be solved in each interval. Needless to say, this procedure to solve the two-point boundary value problem is computation intensive and the convergence is highly dependent on the initial guess of the solution.

In some recent work ([1], [3]), Agrawal and coworkers have proposed a new method for solution of the dynamic optimization problem for classes of linear time-invariant dynamic systems without Lagrange multipliers. They have demonstrated that the dynamic equations can be embedded into the cost functional by a suitable state transformation so that the constrained dynamic optimization problem can be reduced to an unconstrained dynamic optimization problem. Hence, the need for Lagrange multipliers is eliminated. The new optimality equations are no longer first-order differential equations but are very high-order differential equations. They have shown that the resulting high-order differential equations can be solved in a very computation efficient way using weighted-residual methods. This paper extends these results to classes of nonlinear dynamic systems which are feedback linearizable.

The organization of this paper is as follows: Section 2 describes the use of feedback linearization to change a constrained optimization problem into an unconstrained optimization problem. Section 3 describes the new necessary conditions for optimality. In Section 4, the collocation method is used to reduce the differential equations into a set of algebraic equations. These equations are solved numerically using fixed-point iteration. The method is illustrated with examples in Section 5.

2 Nonlinear Transformation

In this section, we seek a globally diffeomorphic transformation $Y = T(X)$ and $U = \alpha(X) + \mathcal{B}(X)V$ such that the new state-space representation for the dynamic system becomes:

$$\dot{Y} = AY + BV \tag{4}$$

where $T(X)$ is in $\mathcal{R}^n$, $\alpha(X)$ and $\mathcal{B}(X)$ are respectively functions in $\mathcal{R}^m$ and $\mathcal{R}^{m\times m}$. In the new state equation, Y and V are respectively in $\mathcal{R}^n$ and $\mathcal{R}^m$. The matrices A and B have the structure

$$A = \begin{bmatrix} 0 & I_m & 0 & \dots & 0 \\ 0 & 0 & I_m & \dots & 0 \\ \vdots & \vdots & \ddots & \ddots & \vdots \\ 0 & 0 & \dots & 0 & I_m \\ 0 & 0 & \dots & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \vdots \\ I_m \end{bmatrix} \tag{5}$$

where I_m is an m-dimensional identity matrix.

Proposition 1 *The nonlinear dynamic system (1) is feedback linearizable to (4), where the forms of A and B are described in (5), if $[G, ad_f G, ..., ad_f^{p-2} G]$ are involutive and $[G, ad_f G, ..., ad_f^{p-1} G]$ constitute a linearly independent set. The diffeomorphism is global if the conditions are satisfied in entire $\mathcal{R}^n$.*

Here, $\mathrm{ad}_f G$ denotes the matrix formed by Lie bracket of $f(X)$ with individual columns of $G(X)$. The symbol $\mathrm{ad}_f^k g_i$ is a recursive definition for the Lie bracket of $f(X)$ with $\mathrm{ad}_f^{k-1} g_i$. The proof of Proposition 1 is given in Appendix.

From the new linear form of the dynamic system, the control vector V and the state vector Y can be written as higher derivatives of Y_1, an m-dimensional vector consisting of the entries $[y_1, y_2, ..., y_m]^T$ from the vector Y.

$$\begin{bmatrix} Y_1 \\ Y_2 \\ \vdots \\ Y_p \\ V \end{bmatrix} = \begin{bmatrix} I_m & 0 & \dots & 0 & 0 \\ 0 & I_m & \dots & 0 & 0 \\ \vdots & \vdots & \ddots & \ddots & 0 \\ 0 & 0 & \dots & I_m & 0 \\ 0 & 0 & \dots & 0 & I_m \end{bmatrix} \begin{bmatrix} Y_1 \\ \dot{Y}_1 \\ \vdots \\ \overset{\dots}{Y}_1^{p-1} \\ \overset{\dots}{Y}_1^{p} \end{bmatrix} \tag{6}$$

From the given boundary conditions of X at t_0 and t_f, the boundary conditions on Y can be computed using the mapping $Y = T(X)$. Subsequently, using Eq. (6), the boundary values of Y_1, $\dot{Y}_1$,..., $\overset{\dots}{Y}_1^{p-1}$ at t_0 and t_f can be computed.

3 Optimality Conditions

On substituting $X = T^{-1}(Y)$, $U = \alpha(T^{-1}(Y)) + \mathcal{B}(T^{-1}(Y))V$ in Eq. (2) and making use of Eq. (6), the cost functional (3) can be rewritten as

$$J = \int_{t_0}^{t_f} F(Y_1, \dot{Y}_1, \ddot{Y}_1, ..., \overset{\dots}{Y}_1^{p}) dt \tag{7}$$

From the principles of variational calculus [7],

$$\begin{aligned} \delta J \;=\; & \int_{t_0}^{t_f} h^T [\tfrac{\partial F}{\partial Y_1} - \tfrac{d}{dt}\tfrac{\partial F}{\partial \dot{Y}_1} + ... + (-1)^p \tfrac{d^p}{dt^p}\tfrac{\partial F}{\partial \overset{\dots}{Y}_1^{p}}] dt \\ & + h^T [\tfrac{\partial F}{\partial \dot{Y}_1} - \tfrac{d}{dt}\tfrac{\partial F}{\partial \ddot{Y}_1} + ... + (-1)^{p-1} \tfrac{d^{p-1}}{dt^{p-1}}\tfrac{\partial F}{\partial \overset{\dots}{Y}_1^{p}}]_{t_0}^{t_f} \\ & + \dot{h}^T [\tfrac{\partial F}{\partial \ddot{Y}_1} - \tfrac{d}{dt}\tfrac{\partial F}{\partial \overset{\dots}{Y}_1} + ... + (-1)^{p-2} \tfrac{d^{p-2}}{dt^{p-2}}\tfrac{\partial F}{\partial \overset{\dots}{Y}_1^{p}}]_{t_0}^{t_f} \\ & + ... + \overset{\dots}{h}^{p-1 T} [\tfrac{\partial F}{\partial \overset{\dots}{Y}_1^{p}}]_{t_0}^{t_f} \end{aligned} \tag{8}$$

where $h = \delta Y_1$, $\dot{h} = \delta \dot{Y}_1$, etc. are respectively the variation of Y_1 and its higher derivatives. From Section 2, the boundary values of Y_1, $\dot{Y}_1$, $\ddot{Y}_1$,..., $\overset{...p-1}{Y_1}$ at t_0 and t_f are known. As a result, h, $\dot{h}$, $\ddot{h}$,..., $\overset{...p-1}{h}$ at t_0 and t_f are zero. Hence, the necessary conditions for optimality with fixed states at t_0 and t_f become

$$\frac{\partial F}{\partial Y_1} - \frac{d}{dt}\frac{\partial F}{\partial \dot{Y}_1} + ... + (-1)^p \frac{d^p}{dt^p}\frac{\partial F}{\partial \overset{...p}{Y_1}} = 0 \qquad (9)$$

The partial derivatives in Eq. (9) can be evaluated to find the optimality equation. If $\bar{F}$ has a positive bilinear form in U, the optimality equations can be demonstrated to have the following form:

$$\overset{...2p}{Y_1} = h(Y_1, \dot{Y}_1, ..., \overset{...2p-1}{Y_1}), \quad h \in \mathcal{R}^m \qquad (10)$$

In summary, a feedback linearizable system with n states and m inputs ($n = pm$), the optimal solution must satisfy m $2p$-order nonlinear differential equation. This result extends a similar result derived for linear time-invariant dynamic systems [1]. These $2p$-order optimality equations are free of Lagrange multipliers. The $2n$ boundary conditions required in the solution are the computed boundary values of Y_1, $\dot{Y}_1$,..., $\overset{...p-1}{Y_1}$ at t_0 and t_f.

Once Eqs. (10) are solved for $Y_1(t)$, the trajectories of $Y(t)$ and $V(t)$ could be computed using Eq. (6). Further, using $X = T^{-1}(Y)$ and $U = \alpha(T^{-1}(Y)) + \mathcal{B}(T^{-1}(Y))V$, the trajectories of $X(t)$ and $U(t)$ can be determined.

4 Numerical Solution

The optimality equations (10), with the boundary conditions, are solved using collocation method. The solution of the problem is selected to lie within the class of admissible functions [6]:

$$Y_1(t) = \Phi_0(t) + \sum_{i=1}^{k} E_i \phi_i(t) \qquad (11)$$

where $\Phi_0(t)$ is an m-dimensional vector of mode functions that satisfies the boundary conditions of Y_1 and its higher derivatives at time t_0 and t_f. $\phi_i(t)$ are mode functions chosen to have derivatives upto $p-1$ zero at the two end time. E_1,...,E_k are m-dimensional unknown vectors which are to be determined. Under these conditions, $Y_1(t)$ satisfies the boundary conditions of the problem regardless of the numerical values of E_1,...,E_k.

In this paper, $\Phi_0(t)$ is chosen to be the following polynomial function of time

$$\Phi_0(t) = \sum_{l=0}^{p} F_l t^l + \sum_{l=1}^{p-1} F_{p+l} t^p (t - t_f)^l \qquad (12)$$

assuming $t_0 = 0$. The coefficients F_l are vectors of dimension m which are solved using the given boundary conditions of Y_1 and its derivatives at t_0 and t_f. With this choice of $\Phi_0(t)$, it has been shown that F_0, F_1,...,F_{2p-1} can be computed without matrix inversions [1]. $\phi_i(t)$ are selected as

$$\phi_i(t) = t^{p-1+i}(t - t_f)^p, \quad i = 1, ..., k \qquad (13)$$

These mode functions possess the property of having zero derivatives up to $p-1$ at t_0 and t_f.

4.1 Collocation Technique

On substituting Eq. (11) in Eq. (10), the expression of the resulting error vector $\epsilon(t)$ is:

$$\epsilon(t) = \overset{...2p}{\Phi_0}(t) + \sum_{i=1}^{k} E_i \overset{...2p}{\phi_i}(t) - h(E_1, E_2, ..., E_k, t) \qquad (14)$$

The error vector is made orthogonal to selected weighting functions $\psi(t)$:

$$\int_{t_0}^{t_f} \epsilon(t)\psi_j(t)\, dt = 0 \quad (\forall j = 1, ..., k) \qquad (15)$$

which are chosen to be Dirac delta functions $\delta(t - t_j)$. On substituting Eq. (14) in (15), it reduces to

$$\sum_{i=1}^{k} E_i \overset{...2p}{\phi_i}(t_j) = \bar{f}(E_1, E_2, ..., E_k, t_j), \quad (j = 1, ...k) \qquad (16)$$

This leads to mk nonlinear equations in mk unknowns, the elements of E_1,...,E_k. Eq. (16) can be rewritten in its component scalar equations:

$$\sum_{i=1}^{k} E_{il} \overset{...2p}{\phi_i}(t_j) = \bar{f}_l(E_1, E_2, ..., E_k, t_j) \qquad (17)$$

where $E_i = (E_{i1}\ E_{i2}\ ...E_{in})^T$, $l = 1, ..., m$ and $j = 1, ..., k$. On evaluating Eq. (17) at k time instances during t_0 and t_f,

$$\begin{bmatrix} \overset{...2p}{\phi_1}(t_1) & \overset{...2p}{\phi_2}(t_1) & ... & \overset{...2p}{\phi_k}(t_1) \\ \overset{...2p}{\phi_1}(t_2) & \overset{...2p}{\phi_2}(t_2) & ... & \overset{...2p}{\phi_k}(t_2) \\ \vdots & \vdots & \ddots & \vdots \\ \overset{...2p}{\phi_1}(t_k) & \overset{...2p}{\phi_2}(t_k) & ... & \overset{...2p}{\phi_k}(t_k) \end{bmatrix} \begin{bmatrix} E_{1l} \\ E_{2l} \\ \vdots \\ E_{kl} \end{bmatrix} = \begin{bmatrix} \bar{f}_l(t_1) \\ \bar{f}_l(t_2) \\ \vdots \\ \bar{f}_l(t_k) \end{bmatrix} \qquad (18)$$

From this equation,

$$\begin{bmatrix} E_{1l} \\ E_{2l} \\ \vdots \\ E_{kl} \end{bmatrix} = [G]_k \begin{bmatrix} \bar{f}_l(E_1, E_2, ..., E_k, t_1) \\ \bar{f}_l(E_1, E_2, ..., E_k, t_2) \\ \vdots \\ \bar{f}_l(E_1, E_2, ..., E_k, t_k) \end{bmatrix}, \quad (l = 1, .., m) \qquad (19)$$

where $[G]_k$ is a $(k \times k)$ matrix, which is the inverse of the $(k \times k)$ matrix listed in Eq. (18). Eq. (19) describes a set of mk scalar nonlinear equations from which the nk variables E_{il} are to be determined.

4.2 Numerical Solution

The form of Eq. (19) is typical in solutions by fixed-point iteration schemes [5]. The solution vector at an iteration step $j+1$ is computed using the following procedure:

$$\begin{bmatrix} E_{1l} \\ E_{2l} \\ \vdots \\ E_{kl} \end{bmatrix}^{j+1} = [G]_k^j \begin{bmatrix} f_l(E_1, E_2, ..., E_k, t_1) \\ f_l(E_1, E_2, ..., E_k, t_2) \\ \vdots \\ f_l(E_1, E_2, ..., E_k, t_k) \end{bmatrix}^j, \quad (l = 1, .., n) \tag{20}$$

where j is the iteration counter. It is evident that a convergent solution must possess the following properties: (i) for a given k, the number of modes, the solution of E_1,...,E_k must not change as the iteration counter j is increased to $j+1$, (ii) for a given j, the iteration counter, the solution of E_1,...,E_k must not change as the number of modes k is increased to $k + \Delta k$.

The above computation procedure requires inversion of a $(k \times k)$ matrix for a given k. In this paragraph, it will be shown that as the number of collocation points increases from k to $k + \Delta k$, the computation of $[G]_{k+\Delta k}$ can be done incrementally from $[G]_k$ by inversion of a matrix only of dimension Δk. In order to illustrate this idea, let the $(k \times k)$ matrix of Eq. (18) be denoted by $\mathcal{T}_k$. Then,

$$\mathcal{T}_{k+\Delta k} = \begin{bmatrix} \mathcal{T}_k & \mathcal{T}_{k\ k+\Delta k} \\ \mathcal{T}_{k+\Delta k\ k} & \mathcal{T}_{k+\Delta k\ k+\Delta k} \end{bmatrix} \tag{21}$$

where $\mathcal{T}_{k\ k+\Delta k}$ is the column block and $\mathcal{T}_{k+\Delta k\ k}$ is the row block of $\mathcal{T}_{k+\Delta k}$. This happens because with increase in number of modes from k to $k + \Delta k$, i.e., adding collocation points, the first $(k \times k)$ block within $\mathcal{T}_{k+\Delta k}$ is same as $\mathcal{T}_k$. Now, the following matrix property can be used :

$$[\mathcal{T}_{k+\Delta k}]^{-1} = \begin{bmatrix} [\mathcal{T}_k]^{-1} + \mathcal{Z}_2 \mathcal{Z}_1^{-1} \mathcal{Z}_3 & -\mathcal{Z}_2 \mathcal{Z}_1^{-1} \\ -\mathcal{Z}_1^{-1} \mathcal{Z}_3 & \mathcal{Z}_1^{-1} \end{bmatrix} \tag{22}$$

where

$$\begin{aligned} \mathcal{Z}_1 &= \mathcal{T}_{k+\Delta k\ k+\Delta k} - \mathcal{T}_{k+\Delta k\ k}[\mathcal{T}_k]^{-1}\mathcal{T}_{k\ k+\Delta k} \\ \mathcal{Z}_2 &= [\mathcal{T}_k]^{-1}\mathcal{T}_{k\ k+\Delta k} \\ \mathcal{Z}_3 &= \mathcal{T}_{k+\Delta k\ k}[\mathcal{T}_k]^{-1} \end{aligned} \tag{23}$$

From this discussion, it is clear that the computational performance of this algorithm can be significantly improved by using recursion to compute $[G]_{k+\Delta k}$ from $[G]_k$.

4.3 Computation Flowchart

The steps of the computation are outlined in a flowchart in Fig. 1. From the perspective of real-time trajectory generation, the computations can be divided into (i) off-line, and (ii) on-line. The results of

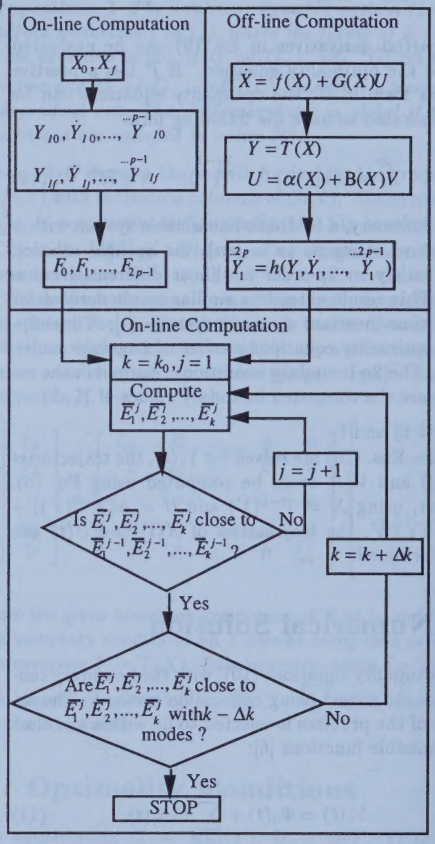

On-line Computation
Off-line Computation
X_0, X_f
$\dot{X} = f(X) + G(X)U$
$Y_{10}, \dot{Y}_{10}, ..., \overset{...p-1}{Y}_{10}$
$Y_{1f}, \dot{Y}_{1f}, ..., \overset{...p-1}{Y}_{1f}$
$Y = T(X)$
$U = \alpha(X) + B(X)V$
$F_0, F_1, ..., F_{2p-1}$
$\overset{2p}{Y}_1 = h(Y_1, \dot{Y}_1, ..., \overset{...2p-1}{Y}_1)$
On-line Computation
$k = k_0, j = 1$
Compute
$\bar{E}_1^j, \bar{E}_2^j, ..., \bar{E}_k^j$
$j = j + 1$
Is $\bar{E}_1^j, \bar{E}_2^j, ..., \bar{E}_k^j$ close to $\bar{E}_1^{j-1}, \bar{E}_2^{j-1}, ..., \bar{E}_k^{j-1}$?
No
$k = k + \Delta k$
Yes
Are $\bar{E}_1^j, \bar{E}_2^j, ..., \bar{E}_k^j$ close to $\bar{E}_1^j, \bar{E}_2^j, ..., \bar{E}_k^j$ with $k - \Delta k$ modes ?
No
Yes
STOP

Figure 1: A computational flowchart for solution of the optimal control problem for open-chain mechanical systems using fourth-order optimality equations.

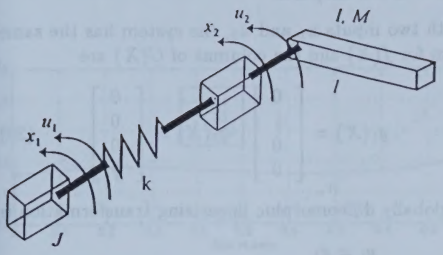


Figure 2: A Single Link Robot with Joint Flexibility.

the off-line computations may be available as a symbolic program or stored data. The on-line computations depend on parameters of the desired motion, e.g., t_f, $X(t_f)$ and must be done after the issuance of the command. Among the steps outlined in the flowchart, the $2p$-order optimality equation of Eq. (10) can be computed off-line. The remaining computations in the flowchart must be performed on-line.

From the flowchart, we observe that the new approach presented in this paper could be quite valuable for generation of optimal trajectories in real time. The off-line computations will provide the expression of the right-hand side vector $h(Y_1, \dot{Y}_1, ..., \overset{...2p-1}{Y_1})$ in Eq. (10). Once the final time t_f and the terminal conditions are provided, in real time, the Eq. (20) must be solved. In the steps leading to the solution of $q(t)$, matrix inversion are required for only $(\Delta k \times \Delta k)$ matrices. In the implementations, Δk could be a very small number. Also, $\Phi_0(t)$ provides a good approximation to $Y_1(t)$ since it does satisfy the boundary values of $Y_1(t)$ at t_0 and t_f upto $p-1$ derivatives. As a result, when the iterations are started out with zero guess of E_1,...,E_k, the convergence is achieved rapidly within a few iterations and a small number of collocation points.

In contrast, if the multiple shooting method was to be implemented in real-time, one still needs computation and inversion of $[(2n+m)(N+1)\times(2n+m)(N+1)]$ Jacobian matrices, numerical integration of $2nN$ first-order differential equations, and a good starting guess to converge to the optimal solution. As a result, we believe that the higher-order method proposed in this paper is very efficient in the generation of optimal trajectories of classes nonlinear systems.

5 Illustrative Example

A single link robot with joint flexibility is used as an example [10]. The system is sketched in Figure 2 and it consists of four states and upto two inputs.

5.1 Single Input

With a single input u_1 and $u_2 = 0$, the description is

$$f(X) = \begin{bmatrix} x_2 \\ -\frac{Mgl}{I}\mathrm{Sin}(x_1) - \frac{k}{I}(x_1 - x_2) \\ x_4 \\ \frac{k}{J}(x_1 - x_2) \end{bmatrix}, g_1(X) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J} \end{bmatrix} \tag{24}$$

Using the techniques of feedback linearization, it can be shown that a suitable linearizing transformation is

$$\begin{aligned} y_1 &= x_1 \\ y_2 &= x_2 \\ y_3 &= -\frac{Mgl}{I}\mathrm{Sin}(x_1) - \frac{k}{I}(x_1 - x_3) \\ y_4 &= -\frac{Mgl}{I}\mathrm{Cos}(x_1)x_2 - \frac{k}{I}(x_2 - x_4) \\ u &= \frac{IJ}{k}(v - a(X)) \end{aligned} \tag{25}$$

where the expression for $a(X)$ is available in [10]. This transformation is globally diffeomorphic and the inverse transformation is

$$\begin{aligned} x_1 &= y_1 \\ x_2 &= y_2 \\ x_3 &= y_1 + \frac{I}{k}(y_3 + \frac{Mgl}{I}\mathrm{Sin}(y_1)) \\ x_4 &= y_2 + \frac{I}{k}(y_4 + \frac{Mgl}{I}\mathrm{Cos}(y_1)y_2) \end{aligned} \tag{26}$$

For this example, $n = 4$, $m = 1$, and $p = 4$. It can be shown that the optimality equation is an 8-th order differential equation in y_1 having the form

$$\overset{...8}{y}_1 = h(y_1, \dot{y}_1, ..., \overset{...7}{y}_1) \tag{27}$$

The boundary conditions are the computed values of y_1, $\dot{y}_1$, $\ddot{y}_1$, $\dddot{y}_1$ at t_0 and t_f. The problem was solved with the method described in this paper as well as the multiple shooting method.

In the problem, the parameters in MKS units are chosen as: $M = 0.1$, $g = 9.8$, $l = 0.1$, $I = 2.0$, $J = 1.0$, $k = 1.0$, and $t_f = 1$. In the cost functional $\bar{F} = u_1^2$ and the boundary conditions are $X(0) = (0.03, 0.01, 0.04, 0.05)^T$ and $X(f) = (0.06, 0.02, 0.08, 0.02)^T$.

The optimal trajectories of the states and control input, computed from the procedure described in this paper, are plotted in Figures 3 and 4. On the same plot, the zero mode solution (only $\Phi_0(t)$), the three mode convergent solution, and multiple shooting solution are overlayed. They are all same within the accuracy of the drawing.

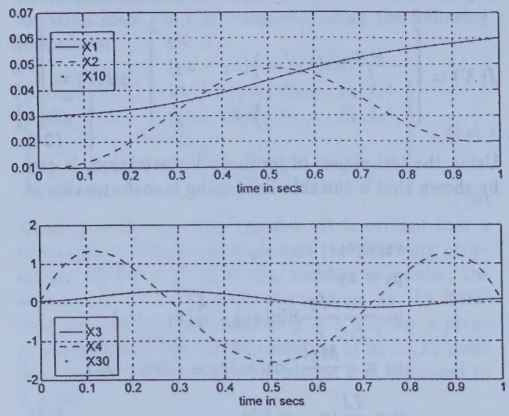


Figure 3: The optimal state trajectories for the example with a single input u_1. The zero mode solution, three mode convergent solution, and multiple shooting solution overlap within the accuracy of drawing.

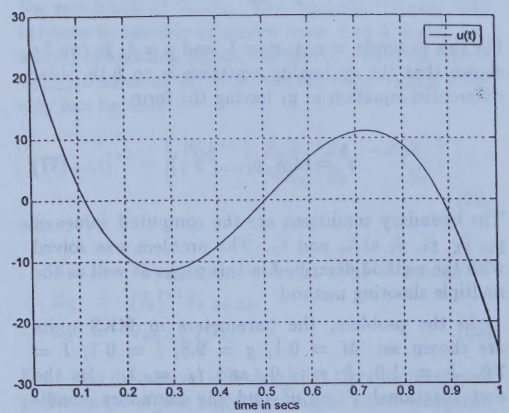


Figure 4: The optimal control trajectories for the example with a single input u_1. The zero mode solution, three mode convergent solution, and multiple shooting solution overlap within the accuracy of drawing

5.2 Two Inputs

With two inputs u_1 and u_2, the system has the same form for $f(X)$ and the columns of $G(X)$ are

$$g_1(X) = \begin{bmatrix} 0 \\ \frac{1}{I} \\ 0 \\ 0 \end{bmatrix}, \quad g_2(X) = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J} \end{bmatrix} \tag{28}$$

A globally diffeomorphic linearizing transformation is

$$\begin{aligned} y_1 &= x_1 \\ y_2 &= x_3 \\ y_3 &= x_2 \\ y_4 &= x_4 \\ u_1 &= -Mgl\mathrm{Sin}x_1 + k(x_1 - x_3) + Iv_1 \\ u_2 &= k(x_3 - x_1) + Jv_2 \end{aligned} \tag{29}$$

With $n = 4$, $m = 2$, and $p = 2$, the optimality equations can be demonstrated to be two 4-th order differential equations in y_1 and y_2. Their forms are

$$\begin{aligned} \ddot{y_1} &= h_1(y_1, y_2, ..., \ddot{y_1}, \ddot{y_2}) \\ \ddot{y_2} &= h_2(y_1, y_2, ..., \ddot{y_1}, \ddot{y_2}) \end{aligned} \tag{30}$$

The boundary conditions are the computed values of y_1, $\dot{y}_1$, y_2, and $\dot{y}_2$ at t_0 and t_f.

For this problem, the parameters in MKS units were chosen as: $M = 0.1$, $g = 9.8$, $l = 0.1$, $I = 2.0$, $J = 1.0$, $k = 1.0$, and $t_f = 1$. The cost functional is $\bar{F} = u_1^2 + u_2^2$ and the boundary conditions are $X(0) = (0.3, 1.0, 0.4, 0.5)^T$ and $X(f) = (0.6, 2.0, 0.8, 2.0)^T$.

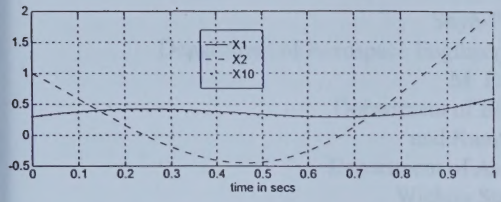
The optimal trajectories of the states and control inputs are plotted in Figures 5 and 6. The zero mode solution, five mode convergent solution, and multiple shooting solution are overlayed on the same plot. They overlap within the accuracy of the drawing.

The procedure described in this paper has been extended to optimization problems with boundary constraints and path constraints [3].

6 Conclusion

A new formulation for dynamic optimization has been proposed for a class of nonlinear systems which is feedback linearizable by a global diffeomorphism. For a system with n states, m control inputs, and an integer $p = \frac{n}{m}$, the new formulation requires solution of m differential equations with highest order as $2p$. In contrast, the conventional approach solves the same problem using $2n$ first-order differential equations in the state, Lagrange multipliers, and control along with m algebraic equations.

The new solution approach has distinct computational advantages for problems where p, the ratio between n and m is large. The reasons are: (i) matrix operations are carried out on $(\Delta k \times \Delta k)$ matrices instead of $(2n+m)(N+1) \times (2n+m)(N+1)$ matrices



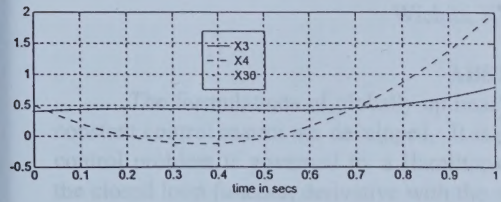


Figure 5: The optimal state trajectories for the example with two inputs u_1 and u_2. The zero mode solution, five mode solution, and multiple shooting solution overlap within the accuracy of drawing.

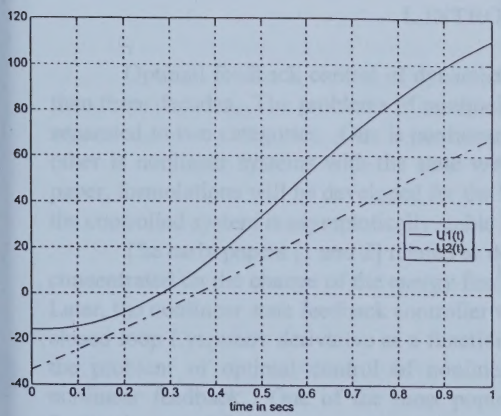


Figure 6: The optimal control trajectories for the example with two inputs u_1 and u_2. The zero mode solution, five mode solution, and multiple shooting solution overlap within the accuracy of drawing.

where Δk could be a small number such as 1, (ii) the knowledge of $2p$ boundary values (up to $p-1$th derivative at the two end time) of $Y_1(t)$ results in the zero mode solution $\Phi_0(t)$ to be a very close approximation of the solution. As a result, the solution is achieved very quickly with a very small number of modes.

7 Acknowledgments

The support of NSF Presidential Faculty Fellowship '94 for the first author is gratefully acknowledged.

8 Appendix

In this section, the proof of Proposition 1 is briefly outlined. It is an extension of the proof for feedback linearization of a single input nonlinear system [10]

Since $Y = T(X)$, $\dot{Y} = \frac{\partial T}{\partial X}(f + \sum_{i=1}^{m} g_i u_i)$. Also $\dot{Y} = AY + BV$, where A and B have the special form of Eq. (5). On equating the expression of $\dot{Y}$, the following result can be derived

$$\bar{T}_{i+1} =< \mathrm{d}\bar{T}_i, f > + \sum_{j=1}^{m} < \mathrm{d}\bar{T}_i, g_j > u_j,$$
$$i = 1, ..., p-1 \qquad (31)$$
$$V =< \mathrm{d}\bar{T}_p, f > + \sum_{j=1}^{m} < \mathrm{d}\bar{T}_p, g_j > u_j \qquad (32)$$

where $\bar{T}_i$ is a m-dimensional vector consisting of the elements $T_{(i-1)m+1}, ..., T_{im}$ and a symbol such as $< \mathrm{d}\bar{T}_i, f >$ denotes the gradient of $\bar{T}_i$ along f. Since L.H.S. of Eq. (31) is independent of u_j, $< \mathrm{d}\bar{T}_i, g_j >= 0$ for $i = 1, ..., p-1$ and $j = 1, ..., m$

Using Jacobi's identity, it can be shown that $\forall j = 1, ..., n$

$$< \mathrm{d}\bar{T}_1, g_j >= 0$$
$$< \mathrm{d}\bar{T}_1, \mathrm{ad}_f g_j >=< \mathrm{d}\bar{T}_2, g_j >= 0$$
$$...$$
$$< \mathrm{d}\bar{T}_1, \mathrm{ad}_f^{p-2} g_j >=< \mathrm{d}\bar{T}_{p-1}, g_j >= 0$$
$$< \mathrm{d}\bar{T}_1, \mathrm{ad}_f^{p-1} g_j >=< \mathrm{d}\bar{T}_p, g_j >\neq 0 \qquad (33)$$

Frobeneus' theorem gaurantees existence of $\bar{T}_1(X)$ only if the set $[G, \mathrm{ad}_f G, ..., \mathrm{ad}_f^{p-2} G]$ is involutive. It can also be demonstrated that if the set $[G, \mathrm{ad}_f G, ..., \mathrm{ad}_f^{p-2} G, \mathrm{ad}_f^{p-1} G]$ is linearly independent, $< \mathrm{d}\bar{T}_1, \mathrm{ad}_f^{p-1} g_j >$ is nonsingular. This procedure provides a construction for $\bar{T}_1(X)$ and using this $\bar{T}_2(X)$,...$\bar{T}_p(X)$ can be formed. This transformation $T(X)$ is globally diffeomorphic if these conditions are met in entire $\mathcal{R}^n$. Once $T(X)$ is established, using Eq. (32), an invertible mapping between U and V is available.

References

[1] Agrawal, S.K. and Veeraklaew, T., "A Higher-Order Method for Dynamic Optimization of a Class of Time-Invariant Linear Systems", *Journal of Dynamic Systems, Measurements, and Control, Transactions of ASME*, December, 1996.

[2] Agrawal, S.K., Li, S., and Fabien, B.C., "Optimal Trajectories of Open-Chain Mechanical Systems: Explicit Optimality Equation with Multiple Shooting Solution", to appear *Mechanics of Structures and Machines*, 1997.

[3] Agrawal, S.K. and Veeraklaew, T., " A New Procedure for Dynamic Optimization of a Class of Linear Systems with Boundary Constraints", In Proceedings, *3rd International Conference on Motion and Vibration Control*, Chiba, Japan, 1996.

[4] Bryson, A. E and Ho, Y.C., *Applied Optimal Control*, Hemisphere Publishing Company, 1975.

[5] Conte, S.D. and de Boor, C., *Elementary Numerical Analysis, An Algorithmic Approach*, McGraw Hill Book Company, 1980.

[6] Fletcher, C.A.J., *Computational Galerkin Methods*, Springer Verlag, 1984.

[7] Gelfand, I.S. and Fomin, S.V., *Calculus of Variations*, Prentice-Hall Book Company, 1963.

[8] Kirk, D.E., *Optimal Control Theory: An Introduction*, Prentice Hall Electrical Engineering Series, 1970.

[9] Pinch, E.R., *Optimal Control and the Calculus of Variations*, Oxford University Press, 1993.

[10] Spong, M. W. and Vidyasagar, M., *Robot Dynamics and Control*, John Wiley and Sons, 1989.

GLOBALLY OPTIMAL FEEDBACK CONTROL OF NONLINEAR SYSTEMS: PART I, THE CONSTANT CONTROL MATRIX

Shyh-Pyng Shue,
Department of Aerospace Engineering, email:shue@twsuvm.uc.twsu.edu
M. E. Sawan,
Department of Electrical Engineering
and Kamran Rokhsaz
Department of Aerospace Engineering
Wichita State University
Wichita, KS 67260-0044

ABSTRACT

The formulations of globally optimal feedback control of nonlinear systems with the constant control matrix are developed. It is shown that the solution of the optimal feedback control problem is governed by a Hamilton-Jacobi-Bellman (HJB) equation. Determining the closed loop function derivative with the state variable as a symmetric matrix form of the state variable reduces the HJB equation to an algebraic Riccati equation along with several other algebraic equations. These equations can be formulated into a series of closed loop Lyapunov equations based on the powers of the state variable of the HJB equation. Adjusting the cost function of the nonlinear system, the positive definite closed loop Lyapunov functions can be found successively from powers of the state variables. From the second method of Lyapunov theory, as long as the cost function and the closed loop Lyapunov function are positive definite, the associated feed-back control system is asymptotically stable in the global sense. This is because the system satisfies the second method of the Lyapunov theory. An example is given to illustrate the proposed theorem.

I. INTRODUCTION

Optimal feedback control of dynamical nonlinear systems has been studied for more than three decades. The problems of nonlinear feedback control of nonlinear systems can be separated to two categories. One is nonlinear systems with constant control coefficients, the other is nonlinear systems with the state variables inside the control coefficients. In this paper, formulations will be developed for the first kind of nonlinear control systems such that the controlled system is asymptotically stable in the global sense.

The early papers [1 and 2] related to nonlinear systems with constant control matrices concentrated on the change of the energy function to the integral state variable cost function. Later, the nonlinear state feedback controller was developed based on modeling the nonlinear closed loop Lyapunov derivative as a function of the state vector [2]. This concept changed the problem of optimal control of nonlinear systems from the linearized controller to nonlinear feedback. One of the most popular methods for computing nonlinear optimal feedback control is the ε parameter expansion method [1, 2, 5]. This method assumes that the higher power terms of the state equation are functions of a small parameter, ε. The closed loop Lyapunov function is modeled by a combination of the state vector with this parameter. This results in a closed loop system with a larger stable region than that of the linearly controlled system. However, globally optimal feedback control of nonlinear systems was still not available at that time. Therefore, semi-globally, and globally optimal feedback control of nonlinear systems attracted great attention [6-10].

An excellent application of optimal feedback control of nonlinear systems appeared in late 1970's [5]. In this paper, the authors showed that nonlinear feedback is required for control of an airplane to avoid stall. Based on this paper, several comparisons and similar publications [11-13] were presented. Some of these used neural networks [12] instead of optimal nonlinear feedback, while others extended the concepts to that of gain scheduling [13]. Reference [13] shows that, the nonlinear feedback controller is necessary to avoid control saturation, and to have a large stable domain.

References [10 and 14] show that three necessary conditions for the globally optimal control of nonlinear systems. These conditions are

a) the cost of the system has to be positive,
b) the closed loop Lyapunov function has to be positive, and
c) the HJB equation must be satisfied.

If the above three constraints are simultaneously satisfied, the controlled system is asymptotically stable in the large, because the second method of the Lyapunov theory is satisfied. Note that the HJB equation is used to ensure that the closed loop Lyapunov derivative with respect to the state variables is negative.

Other methods for optimal control of nonlinear systems can be found in References [15-19]. In Reference [15], a formal power series expansion method is used to solve a nonlinear Hamiltonian system. It is shown that with Carleman linearization, a power series solution for the nonlinear optimal control system can be obtained such that the HJB equation is satisfied. References [16-19] focus on the HJB equation and closed loop Lyapunov equation for the nonlinear optimal control problem. The "viscosity solutions" of the HJB equation is emphasized for the nonlinear time dependent optimal control problem in Reference [16]. It is shown that, with the minimum time function near zero and two uniqueness theorems, a positive solution for the HJB equation can be obtained. Based on the Lyapunov function, an algorithm is presented in Reference [17] to obtain the solution for nonlinear systems. It is shown that the steepest decent method can be used with the Lyapunov function to improve the convergence of the nonlinear system. Reference [18] shows a locally stabilized method for the nonlinear optimal control problem. In this case, with a given special constraint for the system, the negative optimal controller can be determined. The method presented in Reference [19] is focused on a special optimal control problem. Based on the integrand of the cost function, the HJB equation is used with a special state cost to obtain the solution of the HJB from a sequence of Lyapunov function derivatives.

In the current paper, a new method is presented which ensure that the nonlinear cost function and Lyapunov function are positive, while satisfying the HJB equation. Therefore, the globally optimal solution of the nonlinear system is achieved. Note that the final cost is not of interest in this paper.

II. PROBLEM STATEMENTS AND METHODOLOGY

1. PROBLEM STATEMENTS AND PRELIMINARY

Consider a nonlinear system as follows,

$$\dot{x} = f(x) + Bu, \tag{1}$$

where $x \in \mathrm{R}^n$, $u \in \mathrm{R}^m$. The coefficients of $f(x)$ and B have compatible dimensions. Note that B is a constant $n \times m$ matrix. The objective is to find a state feedback controller for the above nonlinear system,

$$u = u(x) \tag{2}$$

such that the following cost function is minimized,

$$J = \frac{1}{2}\int_0^\infty \left[q(x) + u^T R u\right] dt, \tag{3}$$

where $q(x) \geq 0$, and $R > 0$ are symmetric. The following Hamilton-Jacobi-Bellman (HJB) equation is used to find the optimal controller,

$$H = \frac{1}{2}\left[q(x) + u^T R u\right] + V_x(x)\left[f(x) + Bu\right] = 0, \tag{4}$$

where $V(x)$ is the closed loop Lyapunov function. Note that from the above HJB equation,

$$V_x(x)\left[f(x) + Bu\right] = \dot{V}(x) = -\frac{1}{2}\left(q(x) + u^T R u\right) \leq 0, \tag{5}$$

if $q(x) \geq 0$, and $R > 0$, and $H = 0$. If $V(x)$ can be found as a positive definite function, then the above controlled system will satisfy the second method of the Lyapunov theory. Therefore, the solution for the optimal controller will render the system asymptotically stable in the global sense. Note that the above state cost, $q(x)$, does not have to be quadratic, as long as it is positive semi-definite.

2. ASSUMPTIONS AND SIMPLIFICATIONS

Assume the nonlinear state equation can be express as,

$$f(x) = A_1 x + A_2(x) + A_3(x) + \cdots\cdots + A_N(x), \tag{6}$$

where the subscripts of the above equation represent powers of the state variables, and N is the highest power in the state equation. Note that $A_2(x)$, $A_3(x)$, $\cdots$, $A_N(x)$ are $n \times 1$ vectors. Rearrange the above vectors into a matrix form of the state variables as follows,

$$\left.\begin{aligned} A_1 x &= \hat{A}_0 \cdot x \\ A_2(x) &= \hat{A}_1(x) \cdot x \\ A_3(x) &= \hat{A}_2(x) \cdot x \\ &\vdots \quad \vdots \quad \vdots \\ A_N(x) &= A_{N-1}(x) \cdot x \end{aligned}\right\}, \tag{7}$$

where $\hat{A}_0$, $\hat{A}_1(x)$, $\hat{A}_2(x)$ $\cdots$, $\hat{A}_{N-1}(x)$ are $n \times n$ matrices and note that $A_1 = \hat{A}_0$. The above expressions are based on the methods of References [5] and [20]. Therefore, the nonlinear state equation can be expressed in the following vector form,

$$\dot{x} = \begin{bmatrix} \hat{A}_0 \\ \hat{A}_1 \\ \vdots \\ \hat{A}_{N-1} \end{bmatrix}^T \begin{bmatrix} x \\ x \\ \vdots \\ x \end{bmatrix} + Bu \tag{8}$$

Suppose that the closed loop Lyapunov function is a series form of the state variables as follows,

$$V(x) = \begin{cases} \left(\frac{1}{2}x^T P x + \sum_{i=3}^{N+1} V^i(x)\right) & \text{if } N \text{ is odd} \\ \left(\frac{1}{2}x^T P x + \sum_{i=3}^{N+2} V^i(x)\right) & \text{if } N \text{ is even} \end{cases}, \tag{9}$$

where i denotes the power of the state variables. The reason to set up the closed loop Lyapunov function in the above form is that only even powered closed loop Lyapunov

functions can be positive definite. Therefore, the closed loop Lyapunov derivative with respect to the state variables can be obtained as follows,

$$V_x(x) = x^T P + \sum_{i=3}^{N+1} V_x^i(x). \tag{10}$$

Note that from now on, the solution of the closed loop Lyapunov function derivative is derived based on the highest power of the nonlinear state equation being odd. Since P is positive definite and symmetric, the following assumptions are made for the higher powers of the closed loop Lyapunov derivative,

$$\begin{aligned} V_x^2(x) &= x^T \cdot P = x^T \cdot X_0 \\ V_x^3(x) &= x^T \cdot X_1(x) \\ V_x^4(x) &= x^T \cdot X_2(x) \\ &\vdots \quad \vdots \quad \vdots \\ V_x^{N+1}(x) &= x^T \cdot X_{N-1}(x), \end{aligned} \tag{11}$$

where $X_0, X_1, X_2, \cdots, X_{N-1}$ are symmetric $n \times n$ matrices. Note that the even terms of X_i have to be positive definite for the closed loop Lyapunov function to be also positive definite. Therefore, the closed loop Lyapunov derivative can be expressed as,

$$V_x(x) = \begin{bmatrix} x \\ x \\ \vdots \\ x \end{bmatrix}^T \begin{bmatrix} X_0 \\ X_1 \\ \vdots \\ X_{N-1} \end{bmatrix}. \tag{12}$$

Also, the state cost is assumed to be a matrix form of the state variable as,

$$q(x) = \begin{bmatrix} x \\ x \\ x \\ \vdots \\ x \end{bmatrix}^T \begin{bmatrix} Q_0 & \frac{1}{2}Q_1(x) & 0 & \cdots & 0 \\ \frac{1}{2}Q_1(x) & Q_2(x) & \frac{1}{2}Q_3(x) & \ddots & \vdots \\ 0 & \frac{1}{2}Q_3(x) & Q_4(x) & \ddots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & Q_{2N-2}(x) \end{bmatrix} \begin{bmatrix} x \\ x \\ x \\ \vdots \\ x \end{bmatrix}. \tag{13}$$

Note that $q(x)$ is symmetric and positive definite and that the subscripts of the state cost function also represent the powers of the state variables.

The above simplifications and assumptions result in the HJB equation becoming a matrix form of the state variables. Furthermore, the positive definiteness of the closed loop Lyapunov function is easily guaranteed.

3. METHOD OF ANALYSIS

Recall that the nonlinear optimal control problem satisfies the HJB relation in equation (4). This equation can be solved using the Hamiltonian procedure. Following the Hamiltonian stationary condition results in the state feedback optimal controller as follows,

$$u = -R^{-1}B^T V_x^T(x). \tag{14}$$

Substituting equation (14) back into the HJB equation yields the following equality,

$$H = V_x(x)f(x) + \frac{1}{2}q(x) - \frac{1}{2}V_x(x)BR^{-1}B^T V_x^T(x) = 0. \tag{15}$$

Lemma 1: Due to symmetry, the above equation can also be written as,

$$H = \frac{1}{2}\left[V_x(x)f(x) + f^T(x)V_x^T(x) + q(x) - V_x(x)BR^{-1}B^TV_x^T(x)\right] = 0. \quad (16)$$

Proof of *lemma 1*: H is a scalar value, therefore

$$H = H^T = \frac{H + H^T}{2}. \quad (17)$$

Substituting values of H and H^T into the right hand side of equation (17) results in equation (16). Note that $q(x)$ is symmetric, positive semi-definite.

Substituting equations (8), (12), and (13) into equation (16), results in

$$H = \frac{1}{2}\begin{bmatrix} x \\ x \\ \vdots \\ x \end{bmatrix}^T \begin{bmatrix} H_{1,1} & H_{1,2} & \cdots & \\ H_{2,1} & H_{2,2} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \vdots \\ & \cdots & \cdots & H_{N-1,N-1} \end{bmatrix} \begin{bmatrix} x \\ x \\ \vdots \\ x \end{bmatrix} = 0, \quad (18)$$

where

$$H_2 = H_{1,1} = \hat{A}_0^T X_0 + X_0\hat{A}_0 + Q_0 - X_0BR^{-1}B^TX_0 = 0, \quad (19)$$

$$H_3 = H_{1,2} + H_{2,1} = \hat{A}_0^T X_1 + X_1\hat{A}_0 + \hat{A}_1^T X_0 + X_0\hat{A}_1 + Q_1 - X_0BR^{-1}B^TX_1 - X_1BR^{-1}B^TX_0 = 0, \quad (20)$$

$$H_4 = H_{1,3} + H_{3,1} + H_{2,2} = \hat{A}_0^T X_2 + X_2\hat{A}_0 + \hat{A}_1^T X_1 + X_1\hat{A}_1 + \hat{A}_2^T X_0 + X_0\hat{A}_2 + Q_2 - X_0BR^{-1}B^TX_2 - X_2BR^{-1}B^TX_0 - X_1BR^{-1}B^TX_1 = 0, \quad (21)$$

$$\vdots \qquad \vdots \qquad \vdots$$

$$H_{i+2} = H_{1,i+1} + H_{2,i} + H_{i+1,1} = \sum_{j=0}^{i}[\hat{A}_j^T X_{i-j} + X_{i-j}\hat{A}_j] + Q_i - \sum_{j=0}^{i}\left[X_jBR^{-1}B^TX_{i-j}\right] = 0. \quad (22)$$

Note that the above equations are arranged based on the power of the HJB equation. Taking out the first power of the state variables, the above equations become the formulations for $n \times n$ matrices. These equations can be solved successively by substituting the solutions of the lower powers into the expressions for the higher power terms. Since the optimal control input shown in equation (14) is determined from the closed loop Lyapunov function derivatives, the objective of the solution of the HJB equation is to find these derivatives. These parameters can be found from equations (19)-(22) according to the following procedure.

i. <u>The second-power term of the HJB equation</u>

The second-power term of the HJB equation is reduced to an algebraic Riccati equation (ARE) shown in equation (19). The constraints for the solution of this ARE are described as follows [24],

a) $(\hat{A}_0 \quad B)$ is stabilizable,

b) $(\hat{A}_0 \quad D)$ is detectable, where $D^TD = Q_0$, and

c) $R > 0$, and $Q_0 \geq 0$.

If the above three conditions are satisfied, the solution of the ARE will give the system a positive semi-definite symmetric matrix solution. This method is known as the linear quadratic regulator. In addition, the linear closed loop state matrix,

$$A_C = \hat{A}_0 - BR^{-1}B^T X_0, \tag{23}$$

will be Hurwitz which implies a stable linear closed loop state matrix. This also ensures the stability of the nonlinear controlled system in a small around the origin.

ii. The third-power term of the HJB equation

The third-power of the HJB equation is shown in equation (20).

$$H_3 = H_{1,2} + H_{2,1} = \left(\hat{A}_0^T - X_0 BR^{-1}B^T\right)X_1 + X_1\left(\hat{A}_0 - BR^{-1}B^T X_0\right) + \hat{A}_1^T X_0 + X_0\hat{A}_1 + Q_1$$

$$= A_C^T X_1 + X_1 A_C + \hat{Q}_1 = 0 \tag{24}$$

where

$$\hat{Q}_1 = \hat{A}_1^T X_0 + X_0\hat{A}_1 + Q_1. \tag{25}$$

Note that all values in equation (25) are known a priori, except for Q_1. Q_1 is selected such that X_1 is very small or a null matrix. This selection will allow the closed loop Lyapunov function of the system to be positive definite in the next step and facilitates selection for Q_2. Note that equation (24) is a closed loop Lyapunov equation with the-first power of the state variables inside. Therefore, this equation is named the first power Lyapunov equation. Also, since A_C is stable, as long as the system is stabilizable and detectable, X_1 will be positive definite, if $\hat{Q}_1$ is positive definite. However, observability of $(A_C, \hat{Q}_1)$ has to be satisfied.

iii. The fourth-power term of the HJB equation

Following the previous case, the fourth power term of the HJB equation can be written as

$$H_4 = H_{1,3} + H_{2,2} + H_{3,1} = A_C^T X_2 + X_2 A_C + \hat{Q}_2 = 0, \tag{26}$$

where

$$\hat{Q}_2 = \hat{A}_1^T X_1 + X_1\hat{A}_1 + \hat{A}_2^T X_0 + X_0\hat{A}_2 + Q_2 - X_1 BR^{-1}B^T X_1. \tag{27}$$

In this case, Q_2 is chosen to be positive definite while

$$Q = \begin{bmatrix} Q_0 & \frac{1}{2}Q_1 \\ \frac{1}{2}Q_1 & Q_2 \end{bmatrix} \geq 0 \text{ for all } x \neq 0, \text{ and } \hat{Q}_2 > 0. \tag{28}$$

The choice of Q_2 depends on the values of $\hat{A}_1$, and $\hat{A}_2$. Since $\hat{Q}_2$ is positive definite, the second-power closed loop Lyapunov equation will also result in a symmetric positive definite solution for X_2. The above procedure guarantees positive definiteness of the fourth order closed loop Lyapunov function, that is $V^4(x) \geq 0$. Successive application of the above procedure results in solution for all X_i. In every case, Q is chosen such that X_i becomes a small or null matrix for odd values of i, and positive definite for even values of i. Note that Q_i has to be selected such that is positive semi-definite. The above procedure has to be repeated until positive definite $V(x)$, and $q(x)$ are obtained. Therefore, the associated nonlinear system controlled by the closed loop Lyapunov derivatives will be asymptotically stable in the global sense.

iv. The higher-power terms of the HJB equation

The higher-power terms of the HJB equation are shown in equation (22). This equation can be simplified as the ith power Lyapunov equation as follows,

$$H_{i+2} = H_{1,i+1} + H_{2,i} + \cdots + H_{i+1,1} = A_C^T X_i + X_i A_C + \hat{Q}_i , \tag{29}$$

where

$$\hat{Q}_i = \sum_{j=1}^{i} [\hat{A}_j^T X_{i-j} + X_{i-j} \hat{A}_j] + Q_i + \sum_{\substack{j=1,k=1 \\ j+k=i}}^{i-1} X_j B R^{-1} B^T X_k . \tag{30}$$

Note that the above expressions are used for the general cases, and i, j, and $k \neq 0$ are positive integers. As long as the solutions of X_{i-j} are computed successively from the ARE, the values in equation (30) will all be known. In addition, the i^{th} power Lyapunov equation is linear with the i^{th} power of the state variables inside. This equation can be solved easily using standard linear algebraic techniques.

III. RESULTS AND DISCUSSION

1. EXAMPLE

Consider a first order system as follow,

$$\dot{x} = x + x^3 + u \tag{31}$$

with a following cost function

$$J = \frac{1}{2}\int_0^\infty \left\{ \begin{bmatrix} x \\ x \end{bmatrix}^T \begin{bmatrix} q_0(x) & 0 \\ 0 & q_2(x) \end{bmatrix} \begin{bmatrix} x \\ x \end{bmatrix} + u^2 \right\} dt = \frac{1}{2}\int_0^\infty \left\{ \begin{bmatrix} x \\ x \end{bmatrix}^T \begin{bmatrix} 1 & 0 \\ 0 & 8x^2 \end{bmatrix} \begin{bmatrix} x \\ x \end{bmatrix} + u^2 \right\} dt \tag{32}$$

Following the method described in the previous section,

$$X_0 = 1 + \sqrt{2} \text{ and } X_2 = 2\sqrt{2}x^2 \tag{33}$$

Therefore, $A_C = -\sqrt{2}$. The closed loop Lyapunov derivative can be obtained as

$$V_x(x) = [\sqrt{2} + 1]x + [2\sqrt{2}\,]x^3 . \tag{34}$$

The nonlinear controlled system becomes

$$\dot{x} = -\sqrt{2}x - \left[2\sqrt{2} - 1\right]x^3 . \tag{35}$$

It can be shown easily that the closed loop Lyapunov function is positive definite while its time derivative is negative definite. Therefore, the controlled system is asymptotically stable in the global sense, because the second method of Lyapunov theory is satisfied.

2. POSITIVE DEFINITENESS OF A STATE VARIABLE MATRIX

The interpolated terms of the closed loop Lyapunov derivative and the state cost function consist of finite symmetric and square matrices with the state variables inside. If these matrices are positive definite, then the controlled system will have a large stable region. The positive definiteness of a constant matrix can be checked using its eigenvalues, However, if the matrix is square and symmetric, but with the state variables inside, then its positive definiteness has to be checked after some manipulations. Note that only even powers of the state variables in the matrix can guarantee its positive definiteness. If a matrix has odd powers of the state variable inside, then it will be absolutely indefinite.

i. Single variable matrix

In the example of the previous section, the positive definiteness of the state cost function is determined by the interpolated matrix of the state cost function, that is,

$$Q = \begin{bmatrix} q_0 & 0.5q_1(x) \\ 0.5q_1(x) & q_2(x) \end{bmatrix} . \tag{36}$$

The positive definiteness of the above matrix is determined from three conditions

a) $q_0 > 0$,

b) $q_2(x) \geq 0$, and

c) $\det(Q) = q_0 \cdot q_2(x) - \frac{1}{4} q_1(x) \cdot q_1(x) \geq 0$.

It is clear that in the above example, $Q \geq 0$. In this case, Q contains only even powers of the state vector. However, the above three conditions can be equally applied to those cases where odd powers of the state vector are also present in Q.

However, for a symmetric and square matrix, the contribution of the off-diagonal terms is negative. If these terms are small enough, and the diagonal terms are positive definite, the matrix will be positive definite.

ii. Multi-variable matrix

Like the previous simple case, for a multi-variables matrix to be positive definite, several conditions have to be met. These are

a) all diagonal values of the matrix are positive semi-definite for any $x \neq 0$, and

b) all of the sub-matrices are positive definite for any $x \neq 0$, which means,

$$Q_0 \geq 0, \begin{bmatrix} Q_0 & 0.5Q_1 \\ 0.5Q_1 & Q_2 \end{bmatrix} \geq 0, \quad \text{and so on.} \tag{37}$$

In the case, Q_1 has to be small. Therefore, as long as Q_0 and Q_2 are positive semi-definite, the state cost will be positive semi-definite as well.

Note that the selection of Q_2 is based on the degree of the nonlinearity of the state equation. If the nonlinear term appears only in the third power term of the state equation, then a positive difiniteness of the state cost is easily obtained. However, if the state equation has complicated nonlinear powers, the selection of Q_2 for positive definiteness becomes very difficult. In this situation, the following rules can always be used to create a positive definite state variable matrix;

a) even powers of the state variables in the diagonal terms of the matrix have to be made suitably large, and

b) smaller values or zero values must be used in the off diagonal terms of the matrix.

3. THE HJB EQUATION

The HJB equation is constructed from a finite number of the closed loop Lyapunov equations based on the powers of the state variable. The algebraic Riccati equation (ARE) can be considered as the zeroth power closed loop Lyapunov equation, because the ARE can be rewritten as follows,

$$A_C^T X_0 + X_0 A_C + \hat{Q}_0 = 0, \tag{38}$$

where

$$\hat{Q}_0 = Q_0 + X_0 B R^{-1} B^T X_0. \tag{39}$$

$\hat{Q}_i$ is used to compute X_i. From equation (30), $\hat{Q}_i$ is determined from the combination of the lower power open loop Lyapunov equations, the nonlinear state cost, and the terms contributed by the control. Without the nonlinear state cost, Q_i, from the cost function, $\hat{Q}_i$ will not be positive definite with state variables inside. The contribution of the nonlinear state cost function is very significant. When the controlled system cannot be made stable in the large by some linear or nonlinear controller using quadratic cost function, usually proper selection of the nonlinear cost function *will* make the controlled system

asymptotically stable in a large domain with improved performance. However, in some nonlinear systems, state weighting matrices, $\hat{Q}_i$, for the higher power Lyapunov equations do not have to be positive semi-definite. Since the HJB equation gives the controlled system,

$$V_x(x)\,[\,f(x)+Bu\,]=\dot{V}(x)=-\frac{1}{2}[\,q(x)+u^T Ru\,] \\ =-\frac{1}{2}[\,q(x)+V_x(x)BR^{-1}B^T V_x^T(x)\,]\le 0 \qquad (40)$$

the terms contributed by the control, $V_x(x)BR^{-1}B^T V_x(x)$, is unconditionally positive definite. These terms can compensate for the indefinite or negative definite matrices of the nonlinear state cost, $q(x)$. As long as the HJB equation is negative and the closed loop Lyapunov function is positive definite, the system will be asymptotically stable in the large. Like linear cases [21-23], this alternation can generate better performance.

The steps to find the global solution of the nonlinear controlled system do not have to be computed to $i=2N-2$. As long as the closed loop Lyapunov function derivative is fully determined until $i=N$, the system can be controlled with this closed loop Lyapunov function derivative.

IV. CONCLUSIONS

A set of closed loop Lyapunov equations based on the powers of the state variables are shown to provide the solution for the optimal control of nonlinear problems. It is shown that these closed loop Lyapunov equations can be determined from the linear closed loop state matrix. As long as the state cost function can be selected to be positive definite, a positive definite closed loop Lyapunov function can be obtained. Once the cost function and the closed loop Lyapunov function are both positive, the global solution for the optimal control of the nonlinear problem is obtained.

REFERENCES

1. Lukes, D. L., "Optimal Regulation of Nonlinear systems," *SIAM Control and Optimization*, Vol. 7, No. 1, Jan. 1969, pp. 75-100.
2. Willemstein, A. P., "Optimal Regulation of Nonlinear Dynamical Systems on a Finite Interval," *SIAM Control and Optimization*, Vol. 14, No. 6, Nov. 1977, pp. 1051-1069.
3. Garrard, W. L., McClamroch, N. H., and Clark, L. G., "An Approach to Suboptimal Feedback Control of Nonlinear Systems," *International Journal of Control*, Vol. 5, No. 3, 1967, pp. 425-435.
4. Garrard, W. L., "Optimal Regulation of Nonlinear Dynamical Systems," *Automatica*, Vol. 8, No. 4, 1972, pp. 219-221.
5. Garrard, W. L., and Jordan, J. M., "Design of Nonlinear Automatic Flight Control Systems," *Automatica*, Vol. 13, No. 4, 1977, pp. 497-505.
6. Marino, R., and Tomei, P., "Dynamic Output Feedback Linearization and Global Stabilization," *Systems & Control Letters*, Vol. 17, 1991, pp. 115-121.
7. Lin, Z., and Saberi, A., "Semi-global Stabilization of Partially Linear Composite Systems via Feedback of the State of the Linear Part," *Systems & Control Letters*, Vol. 20, No. 2, 1993, pp. 199-207.

8. Khalil, H. K., and Esfandiari, F., "Semiglobal Stabilization of a Class of Nonlinear Systems Using Output Feedback," *IEEE Transactions on Automatic Control,* Vol. Ac-38, No. 9, Sept. 1993, pp. 1412-1415.
9. Battilotti, S., "Remarks on Global Output Feedback Stabilization of a Class of Nonlinear Systems," *Proceedings of the 32nd Conference on Decision and Control,* San Antonio, TX, Dec. 1993, pp. 1948-1953.
10. Mazenc, F., and Praly, L., "Global Stabilization by Output Feedback: Examples and Counterexamples," *Systems & Control Letters,* Vol. 23, No. 1, 1994, pp. 119-125.
11. Garrard, W. L., Enns, D. F., and Snell, S. A., "Nonlinear Feedback Control of Highly Maneuverable Aircraft," *International Journal of Control,* Vol. 56, No. 4, 1992, pp. 799-812.
12. Goh, C. J., "On the Nonlinear Optimal Regulator Problem," *Automatica.* Vol. 29, No. 3, 1993, pp. 751-756.
13. Rounge, P., "Locally Optimal Feedback Strategy for Nonlinear Systems with Uncertain Parameter," *International Journal of Control,* Vol. 47, No. 6, 1988, pp. 1861-1872.
14. Bernstein, D. S., "Nonquadratic Cost and Nonlinear Feedback Control," *Proceedings of the 1991 American Control Conference,* pp. 533-538.
15. Yoshida, T., and Loparo, K A., "Quadratic Regulatory Theory for Analytic Non-linear Systems with Additive Controls," *Automatica*, Vol. 25, No. 4, 1989, pp. 531-544.
16. Evans, L. C., and James, M. R., "The Hamilton-Jacobi-Bellman Equation for Time-Optimal Control," *SIAM Control and Optimization*, Vol. 27, No. 6, Nov. 1989, pp. 1477-1489.
17. Novakovic, Z. R., "Solving Systems of Nonlinear Equations Using the Lyapunov Direct Method," *Computers, Mathematical Applications,* Vol. 20, No. 12, 1990, pp. 19-23.
18. Hermes, H., "Asymptotically Stabilizing Feedback Controls and the Nonlinear Regulator Problem," *SIAM Control and Optimization*, Vol. 28, No. 1, 1990, pp. 185-196.
19. Yanushevsky, R. T., "A Controller for a Class of Nonlinear Systems Using the Lyapunov-Bellman Approach," *Journal of Dynamic Systems, Measurement, and Control,* Vol. 114, No. 3, Sept. 1992, pp. 390-393.
20. Zhang, Y. L. , Gao, J. C., and Zhou, C. H., "Optimal Regulation of Non-linear Systems," *International Journal of Control,* Vol. 50, No. 3, 1989, pp. 993-1000.
21. Johnson, C. D., "The 'Unreachable Pole' Defect in LQR Theory: Analysis and Remedy," *International Journal of Control,* Vol. 47, No. 3, 1988, pp. 697-709.
22. Johnson, C. D., "Limits of Propriety for Linear Quadratic Regulator Problems," *International Journal of Control,* Vol. 45, No. 5, 1987, pp. 1835-1846.
23. Al-Sunni, F. M., Stevens, B. L., and Lewis, F. L., "Negative State Weighting in the Linear Quadratic Regulator for Aircraft Control," *Journal of Guidance, Control and Dynamics*, Vol. 15, No. 5, Sep. 1993, pp. 1279-1281.
24. Anderson, B. D. O., and Moore, J. B., *Optimal Control Linear Quadratic Methods*, Prentice Hall, New Jersey, 07632, 1990.

Relationships in Modern Stochastic Control: Risk-Sensitive and Minimal Cost Variance Control[1]

Chang-Hee Won
TT & C Section, ETRI
Yusong P.O. Box 106
Taejon, Korea 305-600
cwon@kepler.etri.re.kr

Michael K. Sain
Dept. of Electrical Eng.
University of Notre Dame
Notre Dame, IN 46556
sain@krizevac.ee.nd.edu

B. F. Spencer, Jr.
Dept. of Civil Eng. and Geo. Sci.
University of Notre Dame
Notre Dame, IN 46556
spencer@euler.ce.nd.edu

Abstract

Continuing advances in risk-sensitive control have reached a point at which this topic is becoming one of the more intriguing modern paradigms of feedback. Despite a prevailing atmosphere of scrutiny of theoretical studies, risk-sensitive knowledge is growing. Moreover, from the point of view of applications, detailed properties of risk-sensitive design are only beginning to be worked out. The time seems to be right for a survey of the historical underpinnings of the subject. This paper addresses the beginnings and the evolution, over the first quarter-century or so, and points out a close relationship with the notion of optimal cost cumulants, in particular cost variance.

1 Introduction

For a number of years, researchers have been pointing out that the time domain characterization of H_∞ controllers contains a "generalized" Riccati type equation that is also found in linear-quadratic zero-sum differential games and in risk sensitive linear-exponential-quadratic stochastic control. See, for example, [18]. This seems to suggest the possibility of a "grand synthesis" of different approaches to robust control. See Figure 1 for an overview of some of the connections among these various different areas.

The beginning of stochastic control may go all the way back to 1940s when N. Wiener published his results on mean square filtering for weapons fire control developed during World War II. Since then stochastic control went through much theoretical advances by the renowned researchers such as Kalman, Bucy, Wonham, Kushner, Athans, to just name a few [2, 24, 26, 45]. During this time linear quadratic regulator (LQR), linear quadratic Gaussian (LQG) problems have been solved in both full state and output feedback cases. For LQG results refer to excellent texts such as [1, 7, 8, 10, 14]. By the time Doyle and Stein introduced loop transfer recovery (LTR) method in the seventies, stochastic control seemed to have reached its peak [13]. It is around this time that Jacobson came up with the risk-sensitive (RS) idea where the exponential of the quadratic cost function got optimized instead of just the queadratic cost function [21]. This RS idea developed further in the eighties by the researchers such as Speyer, Whittle, and Bensoussan [4, 37, 38, 40]. On the other hand, Minimal Cost Variance(MCV) control, where the idea is to minimize the variance (i.e., the second cumulant) of the cost function instead of the mean, has been developing from the sixties [32, 33, 34, 44]. Currently, RS and MCV control are being actively investigated in the area of stochastic control [6, 16, 43, 44]. Some history

[1]This work was funded in part by the Frank M. Freimann Chair, in the Department of Electrical Engineering at the University of Notre Dame, and by the National Science Foundation under Grants CMS 93-01584 and CMS 95-00301. Submitted for possible presentation at the Thirty-Fourth Annual Allerton Conference on Communication, Control, and Computing. June 28, 1996.

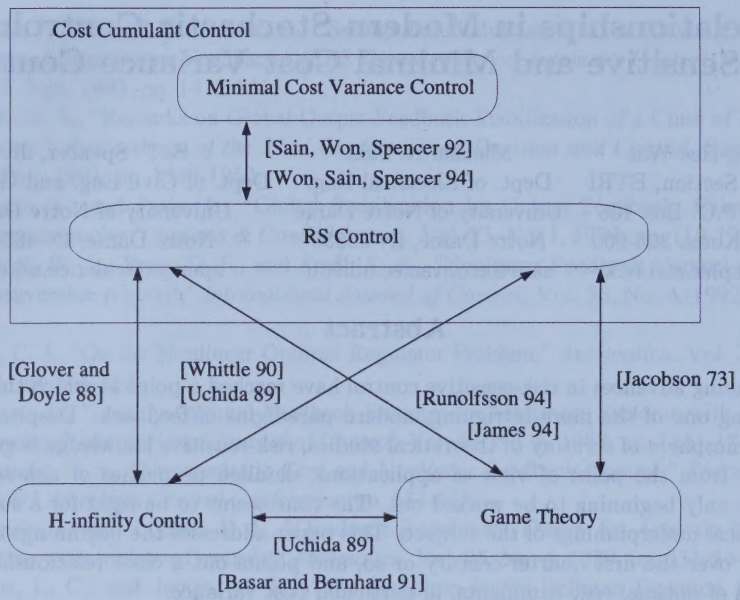


Figure 1: Relations Between Various Robust Controls

and current research in the area of RS and MCV control are discussed in the following sections.

2 A Brief History of Risk-Sensitive Control

Risk-sensitive optimal control seems to have started with Jacobson in 1973. In the 1970s Jacobson extended LQG results by replacing the quadratic criterion with an exponential of a quadratic criterion, and related linear-exponential-quadratic-Gaussian (LEQG) control to differential games [21]. Many years later, Whittle noted Jacobson's results as a risk-sensitive control [42]. Speyer *et al.* [37] extended Jacobson's results to the noisy linear measurements case in discrete time. In [37], optimal control becomes a linear function of the smoothed history of the state, and the solutions are acquired by defining an enlarged state space composed of the entire state history. This enlarged state vector grows at every new stage but retains the feature of being a discrete linear system with additive white Gaussian noise. They also briefly discuss the continuous time terminal LEQG problem in [37], and the solutions are achieved by taking a formal limit of the discrete case solutions. In 1976, Speyer considered the noisy measurement case again in continuous time, but with zero state weighting in the cost function [38]. Unlike the previous work [37], the Hamilton-Jacobi-Bellman equation was used to produce the solutions in [38]. Kumar and van Schuppen derived the general solution of the partially observed exponential-of-integral (EOI) problem in continuous time with zero plant noise in 1981 [25]. Whittle then published his results for the general solution of the partially observed LEOI problem in discrete time [40]. Four years later, Bensoussan and van Schuppen reported the solution to the general case of a continuous time partially observed stochastic EOI problem using a different method from Whittle [4]. Unexpectedly in 1988, Glover and Doyle related H_∞ and minimum entropy criteria to the infinite

horizon version of LEOI theory in discrete time, thus establishing a relationship between Whittle's risk-sensitive control and H_∞ optimal control [18]. This result was extended to continuous time by Glover [19]. In 1990, Whittle published the risk-sensitive maximum principle in book form [41], and published a journal article about the risk-sensitive maximum principle for the case of partially observed states using large deviation ideas [42]. Good references for large deviation theory are [11] and [17]. A couple years later, Bensoussan published a book with all solutions (including the partially observed case) of the exponential-of-integral problem [5]. Başar and Bernhard noted the relationship between deterministic dynamic games and H_∞ optimal control in their book [3]. Also, the relationship between H_2 and H_∞ control has been studied in [9]. In 1992, James states that the risk-sensitive optimal control problem with full-state-feedback information is equivalent to a stochastic differential game problem [22]. Fleming and McEneaney pointed out independent, but similar, results in [15]. In 1993, Won, Sain, and Spencer used Runolfsson's infinite horizon LEOI control results in a structural control application [43]. Hopkins presented discounted EOI solutions in [20]. More recently, in 1994, James, *et al.* published risk-sensitive control and dynamic games solutions for partially observed discrete-time nonlinear systems [22, 23]. Runolfsson presented the relationship between Whittle's risk-sensitive control and stochastic differential games in the infinite-horizon case using large deviation ideas [29]. Most recently, Fleming and McEneaney published infinite-horizon full state feedback RS control results for nonlinear system and nonquadratic cost; and Bensoussan and Eliott published extensions of finite time partially observed RS control results in *SIAM Journal on Control and Optimization*[16, 6].

3 Selected Risk-Sensitive Control Results

The following is a summary of the development of RS control theory. Note that Whittle's RS control cost function is just a logarithmic transformation of the EOI control cost criterion.

- **Jacobson, 1973 [21]**
  - linear-exponential-quadratic-Gaussian (LEQG) problem

$$J = E\left\{\mu \exp(\frac{1}{2}\mu\psi)\right\} \tag{1}$$

$$\psi = \int_{t_0}^{t_F} (x'Qx + u'Ru)\, dt + x'(t_F)Q_F x(t_F) \tag{2}$$

  - μ: $+$ or $-$; completely observed case; continuous and discrete time cases; linear combination of all the moments of ψ

- **Speyer, Deyst, and Jacobson, 1974 [37]**
  - LEQG problem
  - μ: nonzero constant; partially observed case (discrete time case); terminal cost ($Q_i = 0$) problem (discrete and continuous time cases); general problem (discrete time), but controller dependent upon an infinite dimensional state space; no process noise problem (discrete time); no certainty equivalence principle; linear combination of all the moments of ψ

$$J = E\left\{\mu \exp(\frac{1}{2}\mu\psi)\right\} \tag{3}$$

$$\psi = \sum_{i=1}^{N}(x_i'Q_ix_i + u_i'R_iu_i) + x_{N+1}'Q_{N+1}x_{N+1} \tag{4}$$

- **Speyer, 1976 [38]**
  - LEQG problem

$$J = E\left\{\mu \exp(\frac{1}{2}\mu\psi)\right\} \tag{5}$$

$$\psi = \int_{t_0}^{t_F}(u'Ru)\,dt + x'(t_F)Q_Fx(t_F). \tag{6}$$

  - μ: nonzero constant; only terminal state penalized, zero state weighting; partially observed case; continuous time case; no certainty equivalence principle

- **Whittle, 1981 [40]**
  - risk sensitive (RS) problem

$$\gamma(\theta) = -\frac{2}{\theta}\log\left(E\{\exp(-\frac{\theta}{2}\psi)\}\right) \tag{7}$$

$$\psi = \sum_{i=1}^{N-1}(x_i'Qx_i + u_i'Ru_i) + x_N'\Pi_Nx_N \tag{8}$$

  - $(\theta = 0)$ risk neutral case, $(\theta > 0)$ risk seeking case, and $(\theta < 0)$ risk aversive case; completely and partially observed cases; discrete time; new (relative to LQG) certainty equivalence principle; linear combination of cumulants of ψ

- **Kumar and van Schuppen, 1981 [25]**
  - exponential-of-integral (EOI)

$$J = E\left\{\mu \exp(\frac{\mu}{2}\int_0^T c(x,u)\,dt)\right\} \tag{9}$$

  - partially observed case; continuous time; zero plant noise $dx = Fx+Gu$; linear combination of all the moments; [37] candidate for optimality wrong

- **Bensoussan and van Schuppen, 1985 [4]**
  - EOI

$$J = E\left\{\mu \exp\left(\frac{1}{2}\mu\psi\right)\right\} \tag{10}$$

$$\psi = \int_0^{t_F}(x'Qx + u'Nu)\,dt \tag{11}$$

  - linear system, exponential of the quadratic cost; μ: real constant; partially observed; continuous time; method different than Whittle's method [40]

- **Whittle 1991, [42]**

- RS control

$$J = -(k\theta)^{-1} \log E \exp(-k\theta\phi) \tag{12}$$

$$\phi = \int_0^{t_F} c(x,u)dt + \phi_{t_F}(x(t_F)) \tag{13}$$

- θ: real constant; logarithm of exponential-of-integral (LEOI) control; RS maximum principle

- **Bensoussan, 1992 [5]**
  - EOI control

$$J = E\theta \exp(\theta\phi) \tag{14}$$

$$\phi = \int_0^{t_F} c(x,u)dt + \phi_{t_F}(x(t_F)) \tag{15}$$

  - linear system, exponential of the quadratic cost; θ: real constant; full and partial observation cases; continuous time case; nonlinear filtering theory; stochastic maximum principle

- **Runolfsson, 1994 [29]**
  - LEOI control (infinite horizon version of Whittle's RS cost function)

$$J = \lim_{T\to\infty} \frac{1}{T} \log E\{\exp(\mu^2 \int_0^T c(x,u)\ dt)\} \tag{16}$$

  - μ: nonzero constant; solution for infinite horizon, completely observed, risk-aversive; continuous time; relates infinite horizon RS problem to stochastic differential games; no solution for infinite horizon partially observed case

- **James, 1994 [23]**
  - EOI cost function given by

$$J = E\left\{\exp\left(\frac{\mu}{\epsilon}\sum_{j=0}^{M-1} c(x_j^\epsilon, u_j) + \Phi(x_M^\epsilon)\right)\right\} \tag{17}$$

  - finite horizon, partially observed, RS, discrete time, nonlinear, case; relates RS control to the dynamic game problem

- **Fleming and McEneaney, 1995 [16]**
  - LEOI cost fuction given by

$$J = \epsilon \lim_{T\to\infty} \frac{1}{T} \log E\{\exp(\epsilon^{-1} \int_0^T L(x_t, u_t)\ dt)\} \tag{18}$$

  - infinite horizon case; nonlinear system with nonquadratic cost L; full state feedback; continuous time

- **Bensoussan and Elliott, 1995 [6]**
  - extensions of the results in [4, 5]

– dynamics are linear but control parameter may enter both dynamics

$$dx_t = [F(v)x + G(v)]dt + dw_t \tag{19}$$

– exponential of the quadratic cost but control may enter cost nonlinearly; output feedback; continous time

4 A Brief History of Minimal Cost Variance Control

The idea behind cost variance (MCV) control is to minimize the variance of the cost function while keeping the mean at a prespecified level. MCV control may be viewed in a broader context, which we shall call cost cumulant control. In cost cumulant control, linear combinations of cost cumulants are constrained or minimized. The classical minimal mean cost problem is a special case in which the first cumulant is minimized. Minimizing the expected value of cost function was developed in the 1960s; for example see [2, 10, 30, 39]. MCV control was first developed in a dissertation in 1965 [31], and appeared in a journal in 1966 [32]. In 1968, Sain and Souza examined the minimal cost variance concept for problems of estimation[33]. In 1971, Sain and Liberty published an open loop result on minimizing the performance variance while keeping the performance mean at or below a prespecified value [34]. Liberty continued to study characteristic functions of integral quadratic forms, further developing the open loop MCV control idea in a Hilbert space setting [27]. Some years later, Liberty and Hartwig published the results of generating cost cumulants in the time domain [28]. Recently, Sain, Won, and Spencer showed that MCV control is an approximation of risk-sensitive control under some appropriate assumptions [35, 43]. A time-line comparison of MCV control and RS control is shown in Figure 2. Finally, feedback MCV control results are given in [44].

5 Illustrative Feedback Cost Variance Control Result

Consider the Ito sense stochastic diferential equation (SDE) with control,

$$dx(t) = f(t, x(t), u(t))dt + \sigma(t, x(t))dw(t). \tag{20}$$

And the cost function

$$J(t, x(t), k) = \int_t^{t_F} \Big[L(s, x(s), k(s, x(s)))\Big] ds + \psi(x(t_F)). \tag{21}$$

In MCV control we define a class of admissible controllers, then the cost variance is minimized within that class of controllers [36]. Define

$$V_1(t, x; k) = E\{J(t, x(t), k) | x(t) = x\}$$

$$V_2(t, x; k) = E\{J^2(t, x(t), k) | x(t) = x\}.$$

A minimal mean cost control law k_M^* satisfies $V_1(t, x; k_M^*) = V_1^*(t, x) \leq V_1(t, x; k)$, for $t \in T$, $x \in \mathbb{R}^n$, and k an admissible control law. Clearly, $M(t, x) \geq V_1^*(t, x)$.

An MCV control law $k_{V|M}^*$ satisfies $V_2(t, x; k_{V|M}^*) = V_2^*(t, x) \leq V_2(t, x; k)$, for $t \in T$, $x \in \mathbb{R}^n$, whenever k is admissible. The corresponding minimal cost variance is given by $V^*(t, x) = V_2^*(t, x) - M^2(t, x)$ for $t \in T$, $x \in \mathbb{R}^n$.

Here we present the full-state-feedback solution of MCV control for a linear system and a quadratic cost function. For full derivation, refer to [36, 44]. We assume that:

$$\sigma(t, x) = E(t), \tag{22}$$

Minimal Cost Variance Control		*RS Control*	
[Sain 65]	1965		
[Sain 66]	1966		
[Sain and Souza 68]	1968		
[Liberty 71], [Sain and Liberty 71]	1971		
		1973	[Jacobson 73]
		1974	[Speyer, Deyst, Jacobson 74]
[Liberty and Hartwig 76]	1976	1976	[Speyer 76]
[Liberty and Hartwig 78]	1978		
		1981	[Kumar and van Schuppen 81], [Whittle 81]
		1985	[Bensoussan, van Schuppen 85]
		1986	[Whittle and Kuhn 86]
		1990	[Whittle 90]
		1991	[Whittle 91]
[Sain, Won and Spencer 92]	1992		
		1993	[Won, Sain, and Spencer 93]
		1994	[James, Baras, Elliott 94], [Runolfsson 94]
[Won 1995]	1995	1995	[Fleming and McEneany 95]
[Sain, Won, Spencer 95]			[Bensoussan 95]

Figure 2: A Time-Line Comparison of *MCV* and *RS* Control.

$$L(t,x,k(t,x)) = x(t)'Qx(t) + k'(t,x)R(t)k(t,x), \tag{23}$$

$$\psi(x(t_F)) = x'(t_F)Q_F x(t_F), \tag{24}$$

and

$$f(t,x,k(t,x)) = A(t)x(t) + B(t)k(t,x). \tag{25}$$

Then the linear optimal MCV controller is given by

$$k^*_{\mathcal{V}|M}(t,x) = -R^{-1}(t)B'(t)[\mathcal{M}(t) + \gamma(t)\mathcal{V}(t)]x,$$

where $\mathcal{M}$ and $\mathcal{V}$ are the solutions of the coupled Riccati equations (suppressing time):

$$\begin{aligned}\dot{\mathcal{M}} + A'\mathcal{M} + \mathcal{M}A + Q - \mathcal{M}BR^{-1}B'\mathcal{M} \\ +\gamma^2\mathcal{V}BR^{-1}B'\mathcal{V} = 0\end{aligned} \tag{26}$$

and

$$\begin{aligned}\dot{\mathcal{V}} + 4\mathcal{M}EWE'\mathcal{M} + A'\mathcal{V} + \mathcal{V}A - \mathcal{M}BR^{-1}B'\mathcal{V} \\ -\mathcal{V}BR^{-1}B'\mathcal{M} - 2\gamma\mathcal{V}BR^{-1}B'\mathcal{V} = 0,\end{aligned} \tag{27}$$

with boundary conditions $\mathcal{M}(t_F) = Q_F$ and $\mathcal{V}(t_F) = 0$.

References

[1] K. J. Åstrom, *Introduction to Stochastic Control Theory*, New York: Academic Press, 1970.

[2] M. Athans, "The Role and Use of the Stochastic Linear-Quadratic-Gaussian Problem in Control System Design," *IEEE Transactions on Automatic Control*, AC-16, Number 6, pp. 529–551, December 1971.

[3] T. Başar and P. Bernhard, *H_∞–Optimal Control and Related Minimax Design Problems*. Boston: Birkhäuser, 1991.

[4] A. Bensoussan and J. H. van Schuppen, "Optimal Control of Partially Observable Stochastic Systems with an Exponential-of-Integral Performance Index," *SIAM Journal on Control and Optimization*, Volume 23, pp. 599–613, 1985.

[5] A. Bensoussan, *Stochastic Control of Partially Observable Systems*, London: Cambridge University Press, 1992.

[6] A. Bensoussan and R. J. Elliott, "A Finite-Dimensional Risk-Sensitive Control Problem," *SIAM J. Control and Optimization*, Vol. 33, No. 6, November 1995, pp. 1834–1846.

[7] D. P. Bertsekas, *Dynamic Programming and Stochastic Control*. London: Academic Press, 1976.

[8] A. E. Bryson, Jr., and Y.-C. Ho, *Applied and Optimal Control, Optimization, Estimation, and Control*, Revised Printing, New York: Hemisphere Publishing Co., 1975

[9] M. C. Campi and M. R. James, "Risk-Sensitive Control: A Bridge between H_2 and H_∞ Control," *Proceedings of the 32nd Conference on Decision and Control*, San Antonio, Texas, December 1993.

[10] M. H. A. Davis, *Linear Estimation and Stochastic Control*, London: Halsted Press, 1977.

[11] J.-D. Deuschel and D. W. Stroock, *Large Deviations*, San Diego: Academic Press, 1989.

[12] J. L. Doob, *Stochastic Processes*, New York: John Wiley & Sons, Inc., 1953.

[13] J. Doyle and G. Stein, "Robustness with Observers," *IEEE Transactions on Automatic Control*, AC-24, pp. 607–611, August 1979.

[14] W. H. Fleming and R. W. Rishel, *Deterministic and Stochastic Optimal Control*. New York: Springer-Verlag, 1975.

[15] W. H. Fleming and W. M. McEneaney, "Risk Sensitive Optimal Control and Differential Games," *Stochastic Theory and Adaptive Control, Lecture Notes in Control and Information Sciences 184*, T. E. Duncan and B. Pasik-Duncan (Eds.), Springer-Verlag, pp. 185–197, 1992.

[16] W. H. Fleming and W. M. McEneaney, "Risk-Sensitive Control on an Infinite Time Horizon," *SIAM J. Control and Optimization*, Vol. 33, No. 6, November 1995.

[17] M. I. Freidlin and A. D. Wentzell, *Random Perturbations of Dynamical Systems*, New York: Springer-Verlag, 1984.

[18] K. Glover and J. C. Doyle, "State-Space Formulae for All Stabilizing Controllers that Satisfy an H_∞-Norm Bound and Relations to Risk Sensitivity," *Systems and Control Letters*, Volume 11, pp. 167–172, 1988.

[19] K. Glover, "Minimum Entropy and Risk-Sensitive Control: The Continuous Time Case," *Proceedings 28th IEEE Conference on Decision and Control*, pp. 388–391, December 1989.

[20] W. E. Hopkins, Jr., "Exponential Linear Quadratic Optimal Control with Discounting," *IEEE Transactions on Automatic Control*, AC-39, No. 1, pp. 175–179, 1994.

[21] D. H. Jacobson, "Optimal Stochastic Linear Systems with Exponential Performance Criteria and Their Relationship to Deterministic Differential Games," *IEEE Transactions on Automatic Control*, AC-18, pp. 124–131, 1973.

[22] M. R. James, "Asymptotic Analysis of Nonliner Stochastic Risk-Sensitive Control and Differential Games," *Mathematics of Control, Signals, and Systems*, 5, pp. 401–417, 1992.

[23] M. R. James, J. S. Baras, and R. J. Elliott, "Risk-Sensitive Control and Dynamic Games for Partially Observed Discrete-Time Nonlinear Systems," *IEEE Transactions on Automatic Control*, AC-39, No. 4, pp. 780–792, 1994.

[24] R. Kalman, R. Bucy, "New Results in Linear Filtering and Prediction Theory," *ASME Journal of Basic Engineering*, Series D, Vol. 83, pp. 95-108, 1961.

[25] P. R. Kumar and J. H. van Schuppen, "On the Optimal Control of Stochastic Systems with an Exponential-of-Integral Performance Index," *Journal of Mathematical Analysis and Applications*, Volume 80, pp. 312–332, 1981.

[26] H. Kushner *Introduction to Stochastic Control*, New York: Holt,Rinehart and Winston, Inc., 1971.

[27] S. R. Liberty, "Characteristic Functions of LQG Control," Ph.D. Dissertation, Department of Electrical Engineering, University of Notre Dame, August 1971.

[28] S. R. Liberty and R. C. Hartwig, "On the Essential Quadratic Nature of LQG Control-Performance Measure Cumulants," *Information and Control*, Volume 32, Number 3, pp. 276–305, 1976.

[29] T. Runolfsson, "The Equivalence between Infinite-Horizon Optimal Control of Stochastic Systems with Exponential-of-Integral Performance Index and Stochastic Differential Games," *IEEE Transactions on Automatic Control*, Vol. 39, No. 8, pp. 1551–1563, 1994.

[30] A. P. Sage, *Optimum Systems Control.* Englewood Cliffs, NJ: Prentice-Hall Inc., 1968.

[31] M. K. Sain, "On Minimal-Variance Control of Linear Systems with Quadratic Loss," *Ph.D Thesis*, Department of Electrical Engineering and Coordinated Science Laboratory, University of Illinois, Urbana, Illinois, January 1965.

[32] M. K. Sain, "Control of Linear Systems According to the Minimal Variance Criterion—A New Approach to the Disturbance Problem," *IEEE Transactions on Automatic Control*, AC-11, No. 1, pp. 118–122, January 1966.

[33] M. K. Sain and C. R. Souza, "A Theory for Linear Estimators Minimizing the Variance of the Error Squared," *IEEE Transactions on Information Theory*, IT-14, Number 5, pp. 768–770, September 1968.

[34] M. K. Sain and S. R. Liberty, "Performance Measure Densities for a Class of LQG Control Systems," *IEEE Transactions on Automatic Control*, AC-16, Number 5, pp. 431–439, October 1971.

[35] M. K. Sain, Chang-Hee Won, and B. F. Spencer, Jr. "Cumulant Minimization and Robust Control," *Stochastic Theory and Adaptive Control, Lecture Notes in Control and Information Sciences 184*, T. E. Duncan and B. Pasik-Duncan (Eds.), Springer-Verlag, pp. 411-425, 1992.

[36] M. K. Sain, C.-H. Won, and B. F. Spencer, "Cumulants in Risk-Sensitive Control: The Full-State-Feedback Cost Variance Case," *Proceedings of 34th CDC*, New Orleans, 1995.

[37] J. L. Speyer, J. Deyst, and D. H. Jacobson, "Optimization of Stochastic Linear Systems with Additive Measurement and Process Noise Using Exponential Performance Criteria," *IEEE Transactions on Automatic Control*, AC-19, No. 4, pp. 358–366, August 1974.

[38] J. L. Speyer, "An Adaptive Terminal Guidance Scheme Based on an Exponential Cost Criterion with Application to Homing Missile Guidance," *IEEE Transactions on Automatic Control*, AC-21, pp. 371–375, 1976.

[39] F. Tung, "Linear Control Theory Applied to Interplanetary Guidance," *IEEE Transactions on Automatic Control*, AC-9, Number 1, pp. 82–89, January 1964.

[40] P. Whittle, "Risk-Sensitive Linear/Quadratic/Gaussian Control," *Advances in Applied Probability*, Vol. 13, pp. 764–777, 1981.

[41] P. Whittle, *Risk Sensitive Optimal Control.* New York: John Wiley & Sons, 1990.

[42] P. Whittle, "A Risk-Sensitive Maximum Principle: The Case of Imperfect State Observation," *IEEE Transactions on Automatic Control*, Vol. 36, No. 7, pp. 793–801, July 1991.

[43] C.-H. Won, M. Sain, B. Spencer, Jr., "Performance and Stability Characteristics of Risk-Sensitive Controlled Structures Under Seismic Disturbances," *Proceedings American Control Conference*, Seattle, Washington, pp. 1926-1930, June 21, 1995.

[44] C.-H. Won, "Cost Cumulants in Risk-Sensitivity and Minimal Cost Variance Control," Ph. D. Dissertation, University of Notre Dame, Notre Dame, IN, USA, 1995.

[45] W. M. Wonham, "On a Matrix Riccati Equation of Stochastic Control," *SIAM J. Control*, Vol. 6, No. 4, pp. 681–607, 1968.

Robust Control of Markov Decision Processes and Connection to Risk-Sensitive Control

Stefano Coraluppi and Steven Marcus
Electrical Engineering Department and
Institute for Systems Research
University of Maryland
College Park, MD 20742
stefano@isr.umd.edu, marcus@isr.umd.edu

Abstract

This paper introduces a formulation of the robust control problem in the partially observed Markov Decision Process (POMDP) setting. We show that this formulation is the large risk limit of the risk-sensitive control problem. Exploiting this connection, we derive an information state process and dynamic programming equations for the value function. We develop a methodology to determine an optimal policy for finite horizon problems, and near-optimal policies on the infinite horizon. Finally, we introduce an alternative formulation of the robust control problem, leading to stationary policies on the infinite horizon, and provide a methodology to determine optimal policies in this setting.

1 Introduction

Robust control theory has been developed primarily in the linear systems context. It is essentially a minmax approach to control system design, whereby we choose a control to minimize the cost associated with worst-case disturbances. More recently, formulations in the nonlinear setting have been studied. A formulation for Finite State Machines has been proposed in [1]. In this paper, we propose a robust control formulation for MDPs which is precisely the minmax criterion studied in [2]. This formulation does not require much of the machinery in [1], and we believe it is a natural way to formulate the robust control problem in the MDP setting.

The connection between robust and risk-sensitive control in the linear systems setting was established in [6]. It is shown that there is a risk parameter value $\gamma_0 > 0$ such that the solution to the corresponding risk-sensitive control problem yields precisely the optimal H_∞ controller. We show that in the MDP context, the robust control problem is obtained in the limit as $\gamma \to \infty$ of the risk-sensitive problem.

In [1], an information state process and dynamic programming equations are derived for the partially observed, risk-sensitive control problem. We derive an information state process and dynamic programming equations for the robust control problem by taking their limiting form as the risk sensitive parameter tends to infinity. We develop a methodology to determine optimal policies which does not require structural results for the value function.

In general, in the infinite horizon setting there does not exist a stationary optimal policy for the robust control problem. Thus, we consider a finite horizon approximation with sufficiently large horizon, and prove that the resulting policy achieves near-optimality. Additionally, motivated by analogous work in risk-sensitive control (see [8], [3], [7], [4]), we introduce an alternative formulation of the robust control problem for which we show that, on the infinite horizon, there exists a stationary optimal policy. We introduce solution methodologies for this formulation.

2 Problem Formulation

We are interested in discrete-time stochastic dynamical systems with a finite state space and control space. This class of systems can be described by a controlled Markov chain. This is given by $M = (X, Y, U, \{P(u), u \in U\}, \{Q(u), u \in U\})$, where X is the finite state space, Y is the finite output space, and U is the finite set of controls. $P(u)$ and $Q(u)$ are the state transition matrix and the output matrix, respectively, for $u \in U$. That is, $P_{ij}(u) = pr(x_{t+1} = j | x_t = i, u_t = u)$ and $Q_{ij}(u) = pr(y_t = j | x_t = i, u_{t-1} = u)$.

We denote by π_0 the initial distribution of x_0 and define the random variable $C_{k,N} = \sum_{t=k}^{N-1} c_t(x_t, u_t)$. A policy g is a sequence of mappings $g = (g_0, g_1, \ldots)$ such that $u_k = g_k(y^k), y^k = (y_1, \ldots, y_k)$. We denote by G the set of all policies. Given a policy g and an initial distribution π_0, let $p^g_{\pi_0}(\cdot)$ be the corresponding probability distribution on trajectories of the system.

Our objective is to determine the optimal policy or control law with respect to the following cost function:

$$\tilde{J}(g, \pi_0) = \max_{\omega \in \Omega, p^g_{\pi_0}(\omega) \neq 0} C_{0,N}(\omega), \tag{1}$$

where Ω is the set of all trajectories for the system. The maximization is over all possible values that the random variable $C_{0,N}$ takes with nonzero probability. That is, we seek the control law which minimizes the worst case cost incurred.

3 Risk-Sensitive Control

In order to address the task of determining the optimal policy with respect to (1), we will exploit a number of results from risk-sensitive control theory, which we briefly summarize in this section.

Consider the following risk-sensitive cost function:

$$J^\gamma(g, \pi_0) = \frac{1}{\gamma} \log E^g_{\pi_0}[\exp(\gamma C_{0,N})] \tag{2}$$

This cost function is a generalization of the classical stochastic control criterion, which is obtained in the limit $\gamma \to 0$ and which is commonly referred to as a risk-neutral cost. We will consider the following equivalent criterion for $\gamma > 0$:

$$\hat{J}^\gamma(g, \pi_0) = E^g_{\pi_0}[\exp(\gamma C_{0,N})], \gamma > 0. \tag{3}$$

In the fully observed setting, there exists a Markov policy g that is optimal. If the state of the system is partially observed, then we reformulate the problem in terms of an information state. Let $p = |Y|$. A reference measure is defined in [1], under which all observations $y \in Y$ are independent and equiprobable at every time. With this reference measure, which we denote by $\dagger$, we can express the total cost as follows:

$$\begin{aligned} \hat{J}^\gamma(g, \pi_0) &= E^g_{\pi_0}[\exp(\gamma C_{0,N})] \\ &= E^\dagger_{\pi_0}[\lambda_N \exp(\gamma C_{0,N})], \end{aligned} \tag{4}$$

where $\lambda_N := p^N \prod_{t=1}^N Q_{x_t y_t}(u_{t-1})$. Define

$$\sigma_k(i) := E^\dagger_{\pi_0}[\mathbf{1}[x_k = i]\lambda_k \exp(\gamma C_{0,k}) | y_1, \ldots, y_k]. \tag{5}$$

It is shown in [1] that $\sigma_k, k \in \{0, \ldots\}$ constitutes an information state process. It evolves in time as follows:

$$\sigma_0 = \pi_0, \tag{6}$$

$$\sigma_{k+1} = p\sigma_k D^\gamma(k, u_k)\bar{Q}(y_{k+1}, u_k), \tag{7}$$

where $\bar{Q}$ is a diagonal matrix with $\bar{Q}_{ii}(y, u_k) = pr(y_{k+1} = y | x_{k+1} = i, u_k = u)$. There exists a separated policy that is optimal, and the value function is piecewise linear and convex in the information state (see [5]). See [4] for an algorithm to determine optimal policies in this setting.

4 Robust Control

Our first step is to establish that the large risk limit of the risk sensitive control problem is precisely the robust control problem. For this, we need a modified version of the Varadhan-Laplace lemma, which is a standard result in analysis which we state without proof.

Lemma 1 (Varadhan-Laplace). Let F^γ, F be real valued functions defined on a finite set Ω, where $\forall \omega \in \Omega$, $F(\omega) = \lim_{\gamma \to \infty} F^\gamma(\omega)$. Then

$$\lim_{\gamma \to \infty} \frac{1}{\gamma} \log \sum_{\omega \in \Omega} \exp[\gamma F^\gamma(\omega)] = \max_{\omega \in \Omega} F(\omega) \tag{8}$$

Lemma 2 (Modified Varadhan-Laplace). Let F^γ, F be real valued functions defined on a finite set Ω, where $\forall \omega \in \Omega$, $F(\omega) = \lim_{\gamma \to \infty} F^\gamma(\omega)$. Also, let $p(\omega)$ be a nonnegative real $\forall \omega \in \Omega$, independent of γ. Then

$$\lim_{\gamma \to \infty} \frac{1}{\gamma} \log \sum_{\omega \in \Omega} p(\omega) \exp\left[\gamma F^\gamma(\omega)\right] = \max_{\omega \in \Omega, p(\omega) \neq 0} F(\omega) \tag{9}$$

Proof

$$\begin{aligned}
\lim_{\gamma \to \infty} \frac{1}{\gamma} \log \sum_{\omega \in \Omega} p(\omega) \exp\left[\gamma F^\gamma(\omega)\right] &= \lim_{\gamma \to \infty} \frac{1}{\gamma} \log \sum_{\omega \in \Omega, p(\omega) \neq 0} p(\omega) \exp\left[\gamma F^\gamma(\omega)\right] \\
&= \lim_{\gamma \to \infty} \frac{1}{\gamma} \log \sum_{\omega \in \Omega, p(\omega) \neq 0} \exp\left(\gamma [F^\gamma(\omega) + \frac{\log p(\omega)}{\gamma}]\right) \\
&= \max_{\omega \in \Omega, p(\omega) \neq 0} F(\omega)
\end{aligned}$$

by Lemma 1. □

Remark In particular, if $\sum_{\omega \in \Omega} p(\omega) = 1$, i.e. $p(\cdot)$ is a probability measure on Ω, it follows that

$$\lim_{\gamma \to \infty} \frac{1}{\gamma} \log E[\exp\left(\gamma F^\gamma(\omega)\right)] = \max_{\omega \in \Omega, p(\omega) \neq 0} F(\omega) \tag{10}$$

The following theorem establishes the connection between the risk-sensitive control and robust control of MDPs.

Theorem 1 (Cost Function). Let $J^\infty(\cdot, \cdot) = \lim_{\gamma \to \infty} J^\gamma(\cdot, \cdot)$. Let $\tilde{J}(\cdot, \cdot)$ be given by (1). Then $J^\infty = \tilde{J}$.

Proof. Let a policy g and an initial distribution on the states π_0 be given. Let Ω denote the set of possible evolutions of the system, and let $p^g(\omega)$ denote the probability under policy g of the evolution $\omega \in \Omega$.

$$\begin{aligned}
J^\infty(g, \pi_0) &= \lim_{\gamma \to \infty} \frac{1}{\gamma} \log E^g_{\pi_0}[\exp(\gamma C_{0,N})] \\
&= \max_{\omega \in \Omega, p^g_{\pi_0}(\omega) \neq 0} C_{0,N}(\omega)
\end{aligned}$$

using Lemma 2. □

Remark. Note that the dependence of the cost on the state transition matrix and the initial distribution π_0 is limited to whether starting locations and trajectories are feasible or infeasible. That is, for each evolution of the system, its probability $p^g_{\pi_0}(\omega)$ of occuring is relevant only to the extent that it is zero or nonzero.

We are now in a position to derive an information state process and dynamic programming equations for the robust control problem. First, we derive a statistic by taking the large risk limit of a function of the risk-sensitive information state.

The statistic will turn out to be an information state for the MDP with respect to (1).

Theorem 2 (Statistic). Let $\tilde{X}(x', u, k)$ be the set of states at time k from which, using control $u \in U$, there is a nonzero probability that the next state of the system will be x'. Also, let $\tilde{Y}(y, u, k)$ denote the set of states at time k that can result in observation y at time k, if the previous control at time $k-1$ is u.

A statistic for the MDP is given by $\{s_k\}$, $k = 0, 1, \ldots$, where $\forall x,\ x' \in X$,

$$s_0(x) = \begin{cases} 0 & \text{if } \pi_0(x) \neq 0 \\ -\infty & \text{otherwise} \end{cases} \tag{11}$$

$$s_{k+1}(x') = \begin{cases} \max_{x \in \tilde{X}(x', u, k)} [s_k(x) + c_k(x, u)] & \text{if } \tilde{X}(x', u, k) \neq \emptyset \text{ and } x' \in \tilde{Y}(y, u, k+1) \\ -\infty & \text{otherwise} \end{cases} \tag{12}$$

Proof. Let

$$s_k = \lim_{\gamma \to \infty} \frac{1}{\gamma} \log \sigma_k \tag{13}$$

and apply Varadhan-Laplace. Details are omitted for lack of space.

Remark. At time k, if a particular state i is feasible, $s_k(i)$ is the worst case cost incurred in the system's evolution from a feasible starting state to state i at time k. If the state is infeasible, the worst case cost incurred is $-\infty$.

The following result relates the information state and total cost.

Theorem 3 (Cost, Information State).

$$\tilde{J}(g, \pi_0) = \max_{y^N} \max_{i \in X} s_N(i) \tag{14}$$

Proof.

$$\begin{aligned} \tilde{J}(g, \pi_0) &= \lim_{\gamma \to \infty} \frac{1}{\gamma} \log E^{\dagger}_{\pi_0} [\exp(\gamma C_{0,N})] \\ &= \lim_{\gamma \to \infty} \frac{1}{\gamma} \log E^{\dagger}_{\pi_0} [\sum_{i \in X} \sigma_N(i)] \\ &= \lim_{\gamma \to \infty} \frac{1}{\gamma} \log \sum_{y^N} p^N \sum_{i \in X} \sigma_N(i) \\ &= \lim_{\gamma \to \infty} \frac{1}{\gamma} \log \sum_{y^N} \sum_{i \in X} p^N \exp \gamma s_N(i) \\ &= \max_{y^N} \max_{i \in X} s_N(i) \quad \square \end{aligned}$$

We define the value function as

$$Z_{k,N}(s) := \min_{g \in G} \max_{y_{k+1}, \ldots, y_N} \max_{i \in X} s_N(i), \text{ where } s_k = s \tag{15}$$

Thus we have

$$Z_{0,N}(s_0) = Z_{0,N}(\lim_{\gamma \to \infty} \frac{1}{\gamma} \log \pi_0) = \min_{g \in G} \tilde{J}(g, \pi_0) \tag{16}$$

Again by a limiting procedure, we can obtain dynamic programming equations characterizing the value function for the system. First we introduce the following notation for the set of all information states. Define $\tilde{R}_+^{|X|} := \{R_+, -\infty\}^{|X|}$, where R_+ is the set of nonnegative reals.

Theorem 4 (Dynamic Programming). The value function satisfies the following, $\forall s \in \tilde{R}_+^{|X|}$:

$$Z_{N,N}(s) = \max_{i \in X} s(i) \tag{17}$$

$$Z_{k,N}(s) = \min_{u \in U} \max_{y \in Y} Z_{k+1,N}(s_{k+1}(s, u, y)) \tag{18}$$

where $s_{k+1}(s, u, y)$ is given by (12). Furthermore, a policy that achieves the minimum in equations (17) and (18), also achieves the minimum in (15). That is, an optimal separated policy is optimal within the larger class G of all policies.

Proof. Omitted for lack of space.

The fully observed case. In the special case when the state of the system is observed, equations (11),(12) take the following form:

$$s_0(x) = \begin{cases} 0 & \text{if } x = x_0 \\ -\infty & \text{otherwise} \end{cases} \tag{19}$$

$$s_{k+1}(x') = \begin{cases} \max_{x \in \tilde{X}(x', u, k)} [s_k(x) + c_k(x, u)] & \text{if } x' = x_{k+1} \\ -\infty & \text{otherwise} \end{cases} \tag{20}$$

The total cost can be expressed in terms of the information state as follows:

$$\tilde{J}(g, \pi_0) = \max_{i \in X} s_N(i) \tag{21}$$

The value function takes the form

$$Z_{k,N}(s) = \min_{g \in G} \max_{i \in X} s_N(i), \text{ where } s_k = s, \tag{22}$$

with dynamic programming equations given by:

$$Z_{N,N}(s) = \max_{i \in X} s(i) \tag{23}$$

$$Z_{k,N}(s) = \min_{u \in U} Z_{k+1,N}(s_{k+1}(s, u)) \tag{24}$$

The minimization in (24) is unaffected by scaling the information state s. If we normalize the information state, we see that the optimality of a separated policy reduces

to the optimality of a Markov policy, as one would expect. Note that we could have obtained the same results for this special case by directly utilizing the risk-sensitive, fully observed formulation.

Remark. The results in the fully observed case are not new. See [2] for early work on the problem.

For MDPs on an infinite horizon, we consider a formulation with discounted costs. We have

$$\tilde{J}(g, \pi_0) = \sup_{\omega \in \Omega, p^g_{\pi_0}(\omega) \neq 0} C_{0,\infty}(\omega), \tag{25}$$

with $C_{0,\infty}$ given by $C_{0,\infty} = \sum_{k=0}^{\infty} \beta^k c(x_k, u_k), 0 < \beta < 1$. Theorems 1, 2 are valid on the infinite horizon as well. We define the infinite horizon value function as follows:

$$Z_k(s) := \lim_{N \to \infty} Z_{k,N}(s) \tag{26}$$

We can verify that the limit in (26) is well-defined by noting that $\lim_{N\to\infty} S_{k,N}$ is well defined, and $Z_{k,N} = \lim_{\gamma \to \infty} \frac{1}{\gamma} \log S^{\gamma}_{k,N}(\exp(\gamma s))$. Thus

$$Z_k(s) = \lim_{\gamma \to \infty} \frac{1}{\gamma} \log S^{\gamma}_k(\exp(\gamma s)) \tag{27}$$

We can relate the value function to the cost function (25) by taking the limit in (16) as $N \to \infty$. We obtain:

$$Z_0(s_0) = \inf_{g \in G} \tilde{J}(g, \pi_0) \tag{28}$$

The following result characterizes the infinite horizon value function.

Theorem 5. The value function satisfies the following, $\forall s \in \tilde{R}_+^{|X|}$:

$$Z_0(s) = \min_{u \in U} \max_{y \in Y} \beta Z_0\left(\frac{s_1(s, u, y)}{\beta}\right) \tag{29}$$

Proof. Omitted for lack of space.

5 Optimal Robust Policies

In risk-neutral and risk-sensitive control, the determination of optimal policies for partially observed MDPs requires the use of structural results for the value function. Without such results, the minimization over a continuum of information states (the unit simplex), is intractable.

In the robust control setting, the situation is greatly simplified since, on the finite horizon, we need only consider a finite number of information states. At time $k = 0$, there are $2^{|X|} - 1$ values that the information state s_0 can take, corresponding to all

possible subsets of X of feasible initial states. At time $k > 0$, in the worst case there are

$$|s_k| = (2^{|X|} - 1)(|U|\ |Y|)^k \tag{30}$$

feasible states. Note that this number grows exponentially in the size of the horizon.

A simple scheme for determining optimal policies on the finite horizon is the following:

1. Generate all information states of interest.

2. Use the dynamic programming equations (17) and (18) to find the optimal control at each state of interest.

In the special case when the state is fully observed, we can solve (23) and (24) and find a Markov policy that is optimal. Of course, in this case the number of states of interest at each time k remains constant irrespective of the horizon size.

Let us now consider the determination of optimal robust policies on the infinite horizon. Unfortunately, it turns out (as in risk-sensitive control) that in general the optimal policy is nonstationary. In the risk-sensitive setting, this can be understood as a progressive discounting of the risk- sensitive parameter as time increases. In the robust setting, discounted costs affect the optimal choice of control as well.

Our approach to determining an optimal infinite horizon policy will be to approximate the problem by a finite horizon one with sufficiently large horizon, and solving the corresponding optimal control problem. The following result establishes the fact that this methodology gives a policy with a cost that is arbitrarily close to optimal.

Theorem 6. Consider the MDP given by $M = (X, Y, U, \{P(u), u \in U\}, \{Q(u), u \in U\})$, with cost function (25). Let $\epsilon > 0$ be given, and set $N = \max\{\lceil \xi \rceil, 1\}$, where $\xi = \frac{\log[\frac{(1-\beta)\epsilon}{\|c\|}]}{\log \beta}$. Let $\hat{g}$ denote the policy that achieves the minimum in eqns. (17), (18), and is arbitrary but fixed on $(N, \ldots)$. Let $\hat{Z}_k, k = 0, 1, \ldots$ denote the associated value function. Then we have, $\forall s \in \tilde{R}_+^{|X|}$,

$$0 \leq \hat{Z}_0(s) - Z_0(s) \leq \epsilon. \tag{31}$$

Proof. Omitted for lack of space.

6 An Alternate Robust Control Formulation

On the infinite horizon, with discounted costs, our robust control formulation leads to a nonstationary optimal policy in general. This is a somewhat surprising result, and it is shared by the risk-sensitive formulation as well. In the risk-sensitive setting, an interpretation for this is that the discounting of costs amounts to a discounting of the

risk-sensitive parameter. As time increases, the control problem reverts to the risk-neutral one. A formulation which leads to a stationary optimal policy while retaining elements of risk-sensitivity is discussed in [8]. An extension of the formulation in the partially observed setting is introduced in [4].

Motivated by the work in risk-sensitive control, we would like to introduce a formulation for the robust control problem which will lead to a stationary infinite horizon policy, while still preserving aspects of robustness. Specifically, we set the cost-to-go from a state i at time k to be the cost incurred at time k, plus the discounted worst case future cost. In the fully observed setting, this takes the following form:

$$\tilde{h}_{k,N}(i) = \min_{u \in U}\{c_k(i,u) + \beta \max_{j \in \tilde{X}'(i,u)} \tilde{h}_{k+1,N}(j)\} \tag{32}$$

$$\tilde{h}_{N,N}(i) = c_N(i) \tag{33}$$

where $\tilde{X}'(i,u)$ is the set of states that the system reaches in one transition with nonzero probability, given that it is in state i and control u is used. We now show that this formulation is the large risk limit of the risk-sensitive alternate formulation.

Theorem 7. Define $h^{\infty}_{k,N} := \lim_{\gamma \to \infty} h^{\gamma}_{k,N}$, where $h^{\gamma}_{k,N}$ is given by the following:

$$h^{\gamma}_{k,N}(i) = \min_{u \in U}\{c_k(i,u) + \frac{\beta}{\gamma} \log \sum_{j \in X} P_{ij}(u) \exp\left(\gamma h^{\gamma}_{k+1,N}(j)\right)\} \tag{34}$$

$$h^{\gamma}_{N,N}(i) = c_N(i) \tag{35}$$

Then $\tilde{h}_{k,N}(i) = h^{\infty}_{k,N}(i),\ \forall i \in X, 0 \leq k \leq N$.

Proof. Follows by invoking the Modified Varadhan-Laplace Lemma.

In a similar way as in the risk-sensitive formulation, it can be shown that for the infinite horizon with stationary costs, we have $\lim_{N \to \infty} \tilde{h}_{k,N} = \tilde{h}$ independent of k, so that an optimal policy satisfies the following equation, $\forall i \in X$:

$$\tilde{h}(i) = \min_{u \in U}\{c(i,u) + \beta \max_{j \in \tilde{X}'(i,u)} \tilde{h}(j)\} \tag{36}$$

Furthermore, there exists an optimal stationary policy that can be determined through value iteration. At each step of value iteration, the evaluation of the cost associated with a policy can be determined by using the contractive mapping $T^g[z](\cdot)$, where for each $i \in X$,

$$T^g[z](i) = \{c(i,g(i)) + \beta \max_{j \in \tilde{X}'(i,g(i))} z(j)\} \tag{37}$$

In the partially observed setting, this formulation takes the following form, $\forall \pi \in \Pi$, where Π is the $|X|$-dimensional unit simplex.

$$\tilde{h}_{k,N}(\pi) = \min_{u \in U}\{\pi \cdot c_k(u) + \beta \max_{\pi' \in \tilde{X}'(\pi,u)} \tilde{h}_{k+1,N}(\pi')\} \tag{38}$$

$$\tilde{h}_{N,N}(\pi) = \pi \cdot c_N \tag{39}$$

where $\tilde{X}'(\pi, u)$ is the set of information states that the system reaches in one transition with nonzero probability, given that the probability distribution on the current states is given by π, and control u is used.

On the infinite horizon with stationary costs, the partially observed formulation takes the following form:

$$\tilde{h}(\pi) = \min_{u \in U} \{\pi \cdot c(u) + \beta \max_{\pi' \in \tilde{X}'(\pi,u)} \tilde{h}(\pi')\} \tag{40}$$

Structural results are not yet available for the partially observed value function. Thus, both on the finite and infinite horizon, an approximate optimal policy must be determined numerically.

Acknowledgement This research was partially supported by the National Science Foundation under Grant EEC 9402384.

References

[1] J. S. Baras and M. R. James. Robust and risk-sensitive output feedback control for finite state machines and hidden markov models. *J. Math. Systems, Estimation and Control*, to appear.

[2] D. P. Bertsekas and I. B. Rhodes. On the minimax feedback control of uncertain systems. In *Proc. IEEE Conference on Decision and Control*, pages 451–455, 1971.

[3] K. J. Chung and M. J. Sobel. Discounted mdp's: Distribution functions and exponential utility maximization. *SIAM Journal on Control and Optimization*, 25:49–62, 1987.

[4] S. P. Coraluppi and S. I. Marcus. Risk-sensitive control of markov decision processes. In *Proc. 30th Conference on Information Sciences and Systems*, 1996.

[5] E. Fernandez-Gaucherand and S. I. Marcus. Risk-sensitive optimal control of hidden markov models: Structural results. *IEEE Transactions on Automatic Control*, to appear.

[6] K. Glover and J. C. Doyle. State-space formulae for all stabilizing controllers that satisfy an H_∞-norm bound and relations to risk sensitivity. *Systems and Control Letters*, 11:167–172, 1988.

[7] L. P. Hansen and T. J. Sargent. Discounted linear exponential quadratic gaussian control. *IEEE Transactions on Automatic Control*, 40:968–971, 1995.

[8] D. M. Kreps and E. L. Porteus. Temporal resolution of uncertainty and dynamic choice theory. *Econometrica*, 46(1):185–200, 1978.

A SUFFICIENT CONDITION FOR THE STABILITY OF TIME-VARYING STATE-SPACE SYSTEMS: A GENERALIZATION OF THE CAYLEY-HAMILTON THEOREM

Paul R. Bouthellier
Department of Mathematics
Mount Senario College
1500 College Avenue West
Ladysmith, Wisconsin 54848

ABSTRACT

A sufficient condition will be derived under which discrete-time linear time-varying state-space systems will be bounded-input bounded-output stable. The results derived generalize the notion of characteristic equations and the Cayley-Hamilton theorem to the time-varying case. It will be noted that the systems we consider are not required to be slowly varying.

1. INTRODUCTION.

The problem of analyzing the stability of time-varying state-space systems has been studied by many authors [1], [2], [3], [4], [5]. However, outside the cases of the state matrix having slowly varying parameters, being symmetric at each time instance, or being periodic, little is known. Hence in this paper a sufficiency condition will be given under which state-space systems with time-varying system matrices are bounded-input bounded-output (BIBO) stable. The only condition required of the system is that it be uniformly reachable [6], [7] at all times.

In section 2 a state-space transformation given by Hwang [8] is used to transform a given system into a reachable one. It is then shown that a sufficient condition for the stability of a n-th order linear time-varying state-space system is that n n-th order single-input single-output (siso) time-varying difference equations be BIBO stable. The difference equations, which may be viewed as time-varying generalizations of the characteristic equation of the system matrix, have coefficients which are derived using a time-varying version of the Cayley-Hamilton theorem.

In section 3, using the concept of the poles of a time-varying difference equation [9], it will be shown that a sufficient condition for the stability of a n-th order time-varying state-space system is that all n^2 poles of the n characteristic equations derived in section 2 be bounded inside the unit disk. In section 4 concluding remarks will be made.

2. THE STABILITY OF STATE-SPACE SYSTEMS

Consider the n^{th}-order LTV system

$$x(k+1) = A(k)x(k) + B(k)u(k) \tag{2.1}$$

with initial state vector $x(0) \in IR^n$, $A(k) \in IR^{n\times n}$, $B(k) \in IR^{n\times m}$, and $u(k) \in IR^m \; \forall k \in Z$. For now we assume $u(k) = 0 \; \forall k < 0$ and that $m = 1$. The case where the number of inputs m is greater than 1 will be considered later in this section.

In the remainder of this paper the following two assumptions will be made

(A1) System (2.1) is unifomly reachable [6], [7] over Z, i.e. $\forall k \in Z$

$$| \det R(k) | > \epsilon > 0 \tag{2.2}$$

where $R(k)$ is the reachability matrix given by

$$R(k) \equiv [B(k),\, A(k,k)B(k-1),\, A(k,k-1)B(k-2), \ldots, A(k,k-n+2)B(k-n+1)] \tag{2.3}$$

and

$$A(k,k-i) = A(k)A(k-1)A(k-2)\cdots A(k-i) \;\; i \geq 0 \tag{2.4}$$

(A2) The elements of $A(k)$ and $B(k)$ are uniformly bounded functions of time.

We now use a state-space transformation $T(k)$ derived by Hwang [8] which will allow the system (2.1) to be rewritten as a reachable system.

Where $R(k)$ is defined by (2.3) and $\bar{R}(k)$ is defined by

$$\bar{R}(k) \equiv [e_n,\, \bar{A}(k,k)e_n,\, \bar{A}(k,k-1)e_n, \ldots, \bar{A}(k,k-n+2)e_n] \tag{2.5}$$

where e_n is the n-th standard basis vector in IR^n, $\bar{A}(k,k-i)$ is defined similiarly to (2.4), where

$$\bar{A}(k) = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & & \vdots \\ 0 & 0 & 0 & \ldots & 1 \\ -\alpha_0(k) & -\alpha_1(k) & -\alpha_2(k) & \cdots & -\alpha_{n-1}(k) \end{bmatrix} \tag{2.6}$$

and

$$[\alpha_0(k), \alpha_1(k-1), \alpha_2(k-2), \ldots, \alpha_{n-1}(k-n+1)]^T = -R^{-1}(k)A(k,k-n+1)B(k-n) \tag{2.7}$$

we have the following result:

Theorem 2.1[8]: Under the transformation $x(k) = T^{-1}(k)\bar{x}(k)$ where

$$T(k) = \bar{R}(k-1)R^{-1}(k-1) \tag{2.8}$$

the state-space system (2.1) becomes

$$\bar{x}(k+1) = \bar{A}(k)\bar{x}(k) + e_n u(k) \tag{2.9}$$

where $\bar{A}(k)$ is defined by (2.6).

Proof: $T(k)$ and $T^{-1}(k)$ exists as $R(k)$ is invertible by assumption (A1) and $\bar{R}(k)$ is easily shown to be invertible. That

$$T(k+1)A(k)T^{-1}(k) = \bar{A}(k) \text{ and } T(k+1)B(k) = e_n \tag{2.10}$$

can be shown by direct calculation.

That the stability of system (2.9) implies the stability of the original system (2.1) is given by

Theorem 2.2: If the state-space system (2.9) is BIBO stable the system (2.1) is also BIBO stable.

Proof: This follows directly from the fact that the elements of $T^{-1}(k)$ are uniformly bounded functions of time.

We now show that the BIBO stability of a n-th order siso time-varying difference equation is a sufficient condition for the stability of a n^{th} order single-input state-space system of the form (2.1).

Theorem 2.3: If the time-varying difference equation

$$y(k)+\sum_{i=1}^{n} \alpha_{n-i}(k)y(k-i) = u(k), \tag{2.11}$$

where the $\alpha_{n-i}(k)\, i = 1, 2 \ldots, n$ are calculated from (2.7) is BIBO stable then the n^{th} order single-input state-space system (2.1) is BIBO stable.

Proof: It is well-known that the BIBO stability of (2.11) implies the BIBO stability of (2.9). Theorem 2.2 completes the proof.

The stability of time-varying difference equations (2.11) have been studied by several authors (see [5] and [9] and the references contained within).

The stability of state-space systems with m inputs, where m is arbitrary, is now considered.

Reconsider system (2.1) where $B(k) \in IR^{n \times m}$ has uniformly bounded coefficients and $m \geq 1$. If for any bounded scalar inputs $u_i(k) \in Z$

$$x(k+1) = A(k)x(k)+e_i u_i(k) \tag{2.12}$$

are BIBO stable $i = 1, 2, \ldots, n$ (where the e_i are the standard basis vectors in IR^n) system (2.1) is BIBO stable. Define the system given by (2.12) corresponding to the i^{th} basis vector $B(k) = e_i$ byS_i $i = 1, 2, \ldots, n$. So that our earlier work is applicable we shall assume

$$| \det R_i(k) | > \epsilon > 0 \;\; i = 1, 2, \ldots, n \tag{2.13}$$

where

$$R_i(k) = [e_i,\, A(k,k)e_i,\, A(k,k-1)e_i \cdots A(k,k-n+2)e_i] \tag{2.14}$$

$i = 1, 2 \ldots, n$.

Corresponding to each state-space system S_i there corresponds a time-varying difference equation. The stability of these n difference equations will imply the stability of (2.1). The coefficients of the difference equations will be found by using a time-varying version of the Cayley-Hamilton theorem.

To consider the stability of the S_i, we will derive their corresponding characteristic equations. Letting $B(k) = e_i$ (2.7) becomes

$$[e_i,\ A(k,k)e_i,\ A(k,k-1)e_i, \cdots, A(k,k-n+2)e_i] \begin{bmatrix} \alpha_0^i(k) \\ \alpha_1^i(k-1) \\ \vdots \\ \alpha_{n-1}^i(k-n+1) \end{bmatrix} = -A(k,k-n+1)e_i \tag{2.15}$$

where the α's will be superscripted with i to denote that they correspond to the i^{th} system S_i.

Note that (2.15) may be written as

$$A(k,k-n+1)e_i + \sum_{j=1}^{n} \alpha_{n-j}^i(k-n+j)A(k,k-n+1+j)e_i = \underline{0} \tag{2.16}$$

$i = 1, 2, \ldots, n$ where $A(k,k+1) \equiv I_{n \times n}$.

Theorem 2.4: The state-space system S_i is BIBO stable provided the corresponding difference equation

$$y(k) + \sum_{j=1}^{n} \alpha_{n-j}^i(k) y(k-j) = u(k), \tag{2.17}$$

where the $\alpha_j^i(k)\, i = 1, 2, \ldots, n$ satisfy (2.15), is BIBO stable.

Proof: Follows directly from theorem 2.3.

For sake of notation we will define the difference equation given by (2.17) corresponding to the system S_i by p_i.

We now state the main result of this section.

Theorem 2.5: Provided the n time-varying difference equations $p_i\ i = 1, 2, \ldots, n$ are BIBO stable, a n^{th} order state-space system (2.1) with m inputs, $m \geq 1$, is BIBO stable.

Proof: By theorem 2.3 $p_i\, i = 1, 2, \ldots, n$ being BIBO stable imply the $S_i\ i = 1, 2, \ldots, n$ are BIBO stable. This implies that the original system (2.1) is BIBO stable.

Observe that unlike the time-invariant case, where the stability of only 1 polynomial, the characteristic polynomial, need be checked in order to establish the stability of a n-th

order state-space system, in the time-varying case the stability of n difference equations must be established. It will now be shown that the coefficients of the p_i can be found by using a time-varying version of the Cayley-Hamilton theorem.

The n equations (2.16) $i = 1, 2, \ldots, n$ may be combined to give

$$A(k, k-n+1) + A(k, k-n+2) \begin{bmatrix} \alpha_{n-1}^1(k-n+1) & \cdots & 0 \\ 0 & & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \alpha_{n-1}^n(k-n+1) \end{bmatrix} \tag{2.18}$$

$$+A(k, k-n+3) \begin{bmatrix} \alpha_{n-2}^1(k-n+2) & 0 & \cdots & 0 \\ 0 & \alpha_{n-2}^2(k-n+2) & & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_{n-2}^n(k-n+2) \end{bmatrix} + \cdots +$$

$$A(k) \begin{bmatrix} \alpha_1^1(k-1) & 0 & \cdots & 0 \\ 0 & \alpha_1^2(k-1) & & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_1^n(k-1) \end{bmatrix} + I \begin{bmatrix} \alpha_0^1(k) & 0 & \cdots & 0 \\ 0 & \alpha_0^2(k) & & 0 \\ & & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_0^n(k) \end{bmatrix} = O_{n \times n}$$

Using (2.18) we get the following corollary to theorem 2.5

Corollary 2.6: A sufficient condition for a n^{th} order system (2.1) with m inputs, $m \geq 1$, to be BIBO stable is that the n difference equations

$$y(k) + \sum_{j=1}^{n} \alpha_{n-j}^i(k) y(k-j) = u(k) \; i = 1, 2, \ldots, n \tag{2.19}$$

whose coefficients $\alpha_i^j(k)$ are found by solving (2.18), all be BIBO stable.

It will be noted that if the state-space system (2.1) is time-invariant: $A(k) \equiv A$ and $B(k) \equiv B$ (2.18) reduces to

$$A^n + A^{n-1} \begin{bmatrix} \alpha_{n-1}^1(k-n+1) & 0 & \cdots & 0 \\ 0 & \alpha_{n-1}^2(k-n+1) & & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_{n-1}^n(k-n+1) \end{bmatrix} + \cdots +$$

$$A \begin{bmatrix} \alpha_1^1(k-1) & 0 & \cdots & 0 \\ 0 & \alpha_1^2(k-1) & & 0 \\ & & \ddots & \vdots \\ 0 & 0 & \cdots & \alpha_1^n(k-1) \end{bmatrix} + I \begin{bmatrix} \alpha_0^1(k) & 0 & \cdots & 0 \\ 0 & \alpha_0^2(k) & & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \alpha_0^n(k) \end{bmatrix} = O_{n \times n} \tag{2.20}$$

Letting $\alpha_j^i(k) = \alpha_j \, \forall k \in Z$ and $i = 1, 2, \ldots, n$ the n polynomials $p_i \; i = 1, \ldots, n$ all reduce to the same polynomial $y(k) + \sum_{j=1}^{n} \alpha_{n-j} y(k-j) = u(k)$ and corollary 2.6 reduces to:

The system

$$x(k+1) = Ax(k) + Bu(k) \tag{2.21}$$

is BIBO stable if

$$y(k)+\sum_{j=1}^{n}\alpha_{n-j}y(k-j)=u(k) \tag{2.22}$$

is stable where $\alpha_0,\alpha_1,\ldots,\alpha_{n-1}$ satisfy

$$A^n+\alpha_{n-1}A^{n-1}+\alpha_{n-1}A^{n-2}+\cdots+\alpha_0 I=O_{n\times n} \tag{2.23}$$

Hence corollary 2.6 generalizes the well-known time-invariant stability result. Equation (2.18) may be viewed as the time-varying version of the Cayley-Hamilton equation and the n polynomials p_i $i=1,2,\ldots,n$ are time-varying generalizations of the corresponding characteristic equation.

The above results will now be illustrated with an example:

Example 2.7: Consider the period 2 system

$$x(k+1)=A(k)x(k)+B(k)u(k) \tag{2.24}$$

where

$$A(k)=\begin{cases}\begin{bmatrix}.5 & .5\\ .5 & 0\end{bmatrix} & k\text{ even}\\ \begin{bmatrix}0 & .5\\ .5 & .6\end{bmatrix} & k\text{ odd}\end{cases} \tag{2.25}$$

$B(k)\in IR^{2\times m}$, $u(k)\in IR^m$, and the coefficients of $B(k)$ are uniformly bounded functions of time. As $A(k)$ is stable and symmetric $\forall k\in Z$ (2.24) is BIBO stable.

Using (2.15) the two characteristic polynomials p_1 and p_2 corresponding to the second order system (2.24) are given by

p_1
$$y(k)+\alpha_1^1(k)y(k-1)+\alpha_0^1(k)y(k-2)=u(k) \tag{2.26}$$

where

$$\alpha_1^1(k)=\begin{cases}-1.1 & k\text{ even}\\ 0 & k\text{ odd}\end{cases}\quad\text{and}\quad\alpha_0^1(k)=-1/4\ \forall k\in Z \tag{2.27}$$

and

p_2
$$y(k)+\alpha_1^2(k)y(k-1)+\alpha_0^2(k)y(k-2)=u(k) \tag{2.28}$$

where

$$\alpha_1^2(k)=\begin{cases}0 & k\text{ even}\\ -1.1 & k\text{ odd}\end{cases}\quad\text{and}\quad\alpha_0^2(k)=-1/4\,\forall k\in Z \tag{2.29}$$

It is easy to show that (2.26) and (2.28) are BIBO stable. Hence by theorem 2.5 the system (2.24) is also BIBO stable.

It is interesting to note that the period-2 time-varying poles [9] (to be discussed in detail in the next section) of both (2.26) and (2.28) are $\lambda_1 = \lambda_2 = 1/4$; exactly the same as the period-2 eigenvalues of the system matrix for (2.24).

3. THE POLES OF A TIME-VARYING SYSTEMS MATRIX

In this section we shall give a sufficiency condition under which a n^{th} order time-varying system (2.1) is BIBO stable in terms of its time-varying poles. It will be seen that the condition derived generalizes the standard time-invariant stability result.

By theorem 2.5 the n^{th} order system (2.1) is stable provided the n difference equations

$$y(k)+\sum_{j=1}^{n} \alpha_{n-j}^{i}(k)y(k-j) = u(k) \tag{3.1}$$

$i = 1,2,\ldots,n$ defined by (2.17) are BIBO stable.

Conditions under which time-varying difference equations of the form

$$y(k)+\sum_{j=1}^{n} \alpha_{n-j}(k)y(k-j) = u(k) \tag{3.2}$$

are stable will now be given. To discuss the stability of systems of the form (3.2) we will use an operational algebra for such systems developed in [9], [10], [11], [12]. We define two operators as follows: where $\alpha(k) \in IR \forall k \in Z$ define $z^{-l}\bullet$ by $z^{-l} \bullet \alpha(k) = \alpha(k-l)$ and $z^{-l}\circ$ by $z^{-l} \circ \alpha(k) = \alpha(k-l)z^{-l}$. Using these operators (3.2) may be written as

$$(1+\sum_{j=1}^{n} \alpha_{n-j}(k)z^{-j})\bullet y(k) = u(k) \tag{3.3}$$

Factor the time-varying polynomial given by the left-hand side of (3.3) into a product of n 1^{st} order linear factors as follows

$$1+\sum_{j=1}^{n} \alpha_{n-j}(k)z^{-j} = (1-\delta_1(k)z^{-1})\circ(1-\delta_2(k)z^{-1})\circ\cdots\circ(1-\delta_n(k)z^{-1}) \tag{3.4}$$

where $\delta_i(k) \in C\ \forall k \in Z\ i = 1,2,\ldots,n$. The $\delta_i(k)$ are called the time-varying poles [9] of the n-th order polynomial (3.4). A simple algorithm to perform such a factorization is given in [13]. We now relate the stability of time-varying difference equations to their time-varying poles:

Theorem 3.1: A sufficient condition for a difference equation

$$y(k)+\sum_{j=1}^{n} \alpha_{n-j}(k)y(k-j) = u(k) \tag{3.5}$$

to be BIBO stable is that its corresponding polynomial $1 + \sum_{j=1}^{n} \alpha_{n-j}(k)z^{-j}$ admits a factorization as given by (3.4) where the time-varying poles satisfy the condition $|\delta_i(k)| < \delta < 1\ i = 1,2,\ldots,n\ \forall k \in Z$.

For a proof of the above theorem the interested reader is refered to [13].

Note how theorem 3.1 generalizes the standard result for the time-invariant case.

Sufficiency conditions for state-space systems to be stable are now derived.

Define the time-varying poles of the i^{th} characteristic equation (3.1) to be $\delta_j^i(k)$ $j = 1, 2, \ldots, n$, i.e.

$$1+\sum_{j=1}^{n} \alpha_{n-j}^i(k) z^{-j} = (1-\delta_1^i(k)z^{-1}) \circ (1-\delta_2^i(k)z^{-1}) \circ \cdots \circ (1-\delta_n^i(k)z^{-1}) \qquad (3.6)$$

$i = 1, 2, \ldots, n$. The $\delta_j^i(k)$, being the roots of the characteristic equations of $A(k)$, can be thought of as the poles corresponding to $A(k)$. We now state the main result of this section.

Theorem 3.2: A n^{th} order state-space system (2.1) with m inputs, $m \geq 1$, is BIBO stable provided the n^2 poles $\{\delta_j^i(k)\}\, i, j = 1, \ldots, n$ of $A(k)$ satisfy $\mid \delta_j^i(k) \mid < \delta < 1 \, \forall k \in Z$.

Proof: If $\mid \delta_j^i(k) \mid < \delta < 1$ for $j = 1, 2, \ldots, n$, by theorem 3.1 the i^{th} characteristic equation of (2.1) is BIBO stable. Hence $\mid \delta_j^i(k) \mid < \delta < 1 \; \forall i, j = 1, \ldots, n$ imply the n characteristic equations of (2.1) are BIBO stable. Theorem 2.5 completes the proof.

Note that theorem 3.2 generalizes the well-known result from time-invariant system theory: A discrete-time time-invariant state-space system is stable if all the poles of the characteristic equation corresponding to the system matrix lie bounded inside the unit disk.

4. CONCLUSION

In this paper we have given sufficiency conditions under which a time-varying state-space system is stable. The condition that all the (time-varying) poles of the system matrix be bounded inside the unit disk generalizes the time-invariant stability result. The time-varying generalizations of the Cayley-Hamilton theorem, characteristic equations, and the poles of state-space systems have been given. A sequel to this paper will use the results of sections 2 and 3 to derive algorithms for approximating the eigenvalues of time-invariant matrices and to construct new classes of stable time-varying state-space systems.

REFERENCES

[1] H. H. Rosenbock, "Stability of Linear Time-Dependent Control Systems," Journal of Electronics and Control, No. 15, 1963, pp. 73-80.

[2] M. Y. Wu and A. Sherif, "On the Communtative Class of Linear Time-Varying Systems," International Journal of Control, Vol. 23, No. 3, 1976, pp. 433-444.

[3] C.A. Desoer,"Slowly Varying Discrete System $x_{i+1} = A_i x_i$," Electronic Letters, Vol. 6, No. 11, 1970, pp. 339-340.

[4] C.A. Desoer,"Slowly Varying System $\dot{x} = A(t)x$," IEEE Tranactions on Automatic Control, Vol. 14, 1969, pp. 780-781.

[5] P.R. Bouthellier and B.K. Ghosh,"A Stability Theory of Linear Time-Varying Systems," Proc. 25th Annual Allerton Conference on Communication, Control, and Computing, 1987, pp.1172–1180.

[6] D. S. Evans, "Finite-Dimensional Realizations of Discrete-Time Weighting Patterns," SIAM Journal of Applied Mathematics, Vol. 22, No. 1, 1972, pp. 45-67.

[7] L. Weiss, "Controllability, Realization and Stability of Discrete-Time Systems," SIAM Journal of Control and Optimization, Vol. 10, No. 2, 1972, pp. 230-251.

[8] S. K. Hwang, "An Augmentation Approach and Theory of the Resultant for Linear Time-Varying Systems," Ph. D. Dissertation, University of Florida, Gainsville, 1986.

[9] E. W. Kamen, "The Poles and Zeros of a Linear Time-Varying System," Linear Algebra and its Applications, 98: 1988, pp. 263-289.

[10] B.K. Ghosh and P.R. Bouthellier," Simultaneous Coefficient Assignment of Discrete-Time Multi-Input Multi-Output Linear Time-Varying Systems: A New Approach for Compensator Design," SIAM J. Control and Optimization, Vol. 31, No. 6, Nov. 1993, pp. 1438–1461.

[11] E.W. Kamen, P.P. Khargonekar, and K.R. Poolla," A Transfer Function Approach to Linear Time-Varying Discrete-Time Systems," SIAM J. Control and Optimization, Vol. 23, No. 4, July 1985, pp. 550–565.

[12] P.R. Bouthellier and B.K. Ghosh,"Robust Stabilization of Discrete-Time Single-Input Single-Output Linear Time-Varying Systems," Proc. of 27th IEEE Conf. on Decision and Control, 1988, Austin, Texas, pp. 39-44.

[13] P.R. Bouthellier, *Analysis and Design of Discrete-Time Linear Time-Varying Systems,* D.Sc . Dissertation, Washington University, St. Louis, MO, 1990.

A Control Lyapunov Function Approach to Robust Stabilization of Nonlinear Systems*

Marc W. McConley†
Brent D. Appleby‡
Munther A. Dahleh§
Eric Feron¶

October 2, 1996

Abstract

We propose an alternative to gain scheduling for stabilizing a class of nonlinear systems. The computation times required to find stability regions for a given control Lyapunov function vary polynomially with the state dimension for a fixed number of scheduling variables. Control Lyapunov functions to various trim points are used to expand the stability region, and a Lyapunov based synthesis formula yields a control law guaranteeing stability over this region. Robustness to bounded disturbances is easily handled, and the optimal stability margin, defined as a Lyapunov derivative, is recovered asymptotically. We apply the procedure to an example.

1 Introduction

Control of nonlinear systems has been a topic of intense research for some time. Progress on this problem is difficult because of the inherent complexity of general methods which apply to arbitrary systems. An approach which has recently come into favor is the method of *control Lyapunov functions (CLFs)*, which guarantees closed loop stability whenever a CLF can be found. Since no systematic procedure exists for finding CLFs of arbitrary nonlinear systems, research in this direction tends to fall into one of three categories.

- The solution to an optimal control problem based on the solution to a Hamilton-Jacobi equation [1, 7, 18]. It is possible to find approximate solutions to these equations, but this problem is intractable for high order systems.

*This work was supported in part by Charles Stark Draper Laboratory Internal Research and Development, in part by the Air Force Office of Scientific Research under Grant AFOSR F49620-95-0219, and in part by the National Science Foundation under Grants 9157306-ECS and 9409715-ECS.

†M. W. McConley is with the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology and The Charles Stark Draper Laboratory, Inc., 555 Technology Square MS 77, Cambridge, MA 02139.

‡B. D. Appleby is with The Charles Stark Draper Laboratory, Inc., 555 Technology Square MS 77, Cambridge, MA 02139.

§M. A. Dahleh is with the Department of Electrical Engineering, Massachusetts Institute of Technology, Room 35-402, 77 Massachusetts Avenue, Cambridge, MA 02139.

¶E. Feron is with the Department of Aeronautics and Astronautics, Massachusetts Institute of Technology, Room 33-217, 77 Massachusetts Avenue, Cambridge, MA 02139.

- The construction of a stabilizing control law based on a *known* CLF [2, 6, 9, 12, 17]. This does not require the intractable computations of the previous category, but it depends on the existence of a CLF, the construction of which is a separate problem.
- The design of a stabilizing control law for a specific problem or class of nonlinear systems [4, 5, 10, 16]. These approaches can be applied successfully in practical situations, but only for very restricted classes of nonlinear systems.

The main contribution of this paper is a method for solving a version of the nonlinear control problem (Problem 1 below) with the following properties.

- The control Lyapunov functions are computed in a systematic way.
- The computations required are tractable even for high dimensional systems.
- Optimal performance (in the sense of a Lyapunov derivative satisfying a desired stability margin) is recovered asymptotically.
- Robustness to bounded disturbances is handled with only minor modifications.
- The method is applicable to a useful class of nonlinear systems.

We seek a control design to stabilize a nonlinear system in the following sense.

Definition 1 Given a system $\dot{x} = f(x, w)$ with $w(t) \in \mathcal{W} \subseteq \mathbf{R}^l$ for all $t \geq 0$, a positively invariant set $\mathcal{X} \subseteq \mathbf{R}^n$, and a compact subset $\Omega \subset \mathcal{X}$, the system is *robustly uniformly asymptotically stable over* $\mathcal{X}$ *with respect to* Ω, or RUAS($\mathcal{X}$,Ω), if it is uniformly asymptotically stable with respect to Ω (see [9]) whenever $x(0) \in \mathcal{X}$ and $w(t) \in \mathcal{W}$ for all $t \geq 0$. We call the set $\mathcal{X}$ a *region of stability*, or RS, for the system.

In particular, we consider the following control synthesis problem.

Problem 1 Consider a system with control $u(t) \in \mathbf{R}^m$ and disturbance $w(t) \in \mathcal{W} \subseteq \mathbf{R}^l$. The state $x(t) \in \mathbf{R}^n$ is partitioned into $x_N \in \mathbf{R}^k$ and $x_L \in \mathbf{R}^{n-k}$, and the dynamics are

$$\begin{bmatrix} \dot{x}_N \\ \dot{x}_L \end{bmatrix} = \begin{bmatrix} f_N(x_N) \\ f_L(x_N) \end{bmatrix} + \begin{bmatrix} A_N(x_N) \\ A_L(x_N) \end{bmatrix} x_L + \begin{bmatrix} g_w^N(x_N) \\ g_w^L(x_N) \end{bmatrix} w + \begin{bmatrix} g_u^N(x_N) \\ g_u^L(x_N) \end{bmatrix} u, \qquad (1)$$

where $f_N(x_N)$, $f_L(x_N)$, $A_N(x_N)$, and $A_L(x_N)$ are C^1; and $g_w^N(x_N)$, $g_w^L(x_N)$, $g_u^N(x_N)$, and $g_u^L(x_N)$ are continuous. Construct sets $\Omega \subset \mathcal{X} \subseteq \mathbf{R}^n$ containing a desired equilibrium point $x_0 \in \mathbf{R}^n$ and a static state feedback control law $\mu : \mathbf{R}^n \to \mathbf{R}^m$ such that the closed loop system with $u = \mu(x)$ is RUAS($\mathcal{X}$,Ω).

Obviously, we would like $\mathcal{X}$ to be as large as possible and Ω as small as possible. In this paper, we develop a computationally tractable procedure for computing the RS of the system (1), and we use this fact to design a controller to expand the RS.

In many applications, the engineer knows that only a few states affect the system dynamics in a nonlinear way, and the system is modeled in the form (1). Gain scheduling is a common control design approach for this problem. In this method, the "nonlinear" states are treated as scheduling variables, and linear controllers are designed for the system linearized about trim points at fixed values of these variables. The control gains are interpolated based on the scheduling variables. Unfortunately, the system is not guaranteed to be stable when the scheduling variables are changing. The stability properties of such systems are analyzed in [8, 13, 14], but the design of a gain scheduled controller to guarantee stability remains an open problem. The solution in this paper uses *robust control Lyapunov functions* for the system linearized about various trim points to guarantee stability over a range of operating conditions.

2 Main Solution Procedure

Before proceeding to develop the main solution procedure, we define some relevant terms pertaining to systems of the general form

$$\dot{x} = f(x) + g_w(x)w + g_u(x)u, \tag{2}$$

where $f \in C^1(\mathbf{R}^n \to \mathbf{R}^n)$ and $g_w(x)$ and $g_u(x)$ are continuous.

Definition 2 Given a point $x_0 \in \mathbf{R}^n$, a function $V : \mathbf{R}^n \to \mathbf{R}$ is a *positive definite function centered at* x_0 if $V(x_0) = 0$ and $V(x) > 0$ for all $x \neq x_0$.

Definition 3 A *level set* of a proper, positive definite function $V(x)$ is defined by a real number $c \geq 0$ (the corresponding *level value*) via $\Omega \doteq V^{-1}[0, c] = \{x \in \mathbf{R}^n \mid V(x) \leq c\}$.

Definition 4 A point $x_0 \in \mathbf{R}^n$ is a *trim point* of the system (2) if there exists a control $u_0 \in \mathbf{R}^m$ such that $f(x_0) + g_u(x_0)u_0 = 0$. A connected set of trim points is a *trim surface.*

Definition 5 ([6, 15]) Consider a subset $\mathcal{W} \subseteq \mathbf{R}^l$, a point $x_0 \in \mathbf{R}^n$, a positive definite function $W(x)$ centered at x_0, and real numbers $c_2 > c_1 \geq 0$. A locally Lipschitz, proper, positive definite function $V(x)$ centered at x_0 is a *robust control Lyapunov function (RCLF) to* x_0 *with stability margin* $W(x)$ *over* $V^{-1}[c_1, c_2]$ for the system (2) if

$$\sup_{x \in V^{-1}[c_1,c_2] \cap \ker(L_{g_u} V)} \; \sup_{w \in \mathcal{W}} L_f V(x) + L_{g_w} V(x) w + W(x) \leq 0, \tag{3}$$

where $L_h V(x)$ is the Lie derivative of $V(x)$ along the vector field $h(x)$.

A given RCLF decreases along trajectories of the closed loop system under the *inverse optimal control law* [6] whenever $w(t) \in \mathcal{W}$. Therefore, the system is RUAS(Ω_2,Ω_1) for $\Omega_2 \doteq V^{-1}[0, c_2]$ and $\Omega_1 \doteq V^{-1}[0, c_1]$, and Problem 1 reduces to finding an RCLF. First, we design an RCLF to the equilibrium point x_0. We can always do this if the linearized dynamics about x_0 are stabilizable and the disturbances are sufficiently small. Computing an RS based on this RCLF is tractable for the system (1). We design RCLFs to other trim points in the same manner. Trajectories starting in a level set Ω_2^i about x_i converge to a smaller level set $\Omega_1^i \subset \Omega_2^i$. The union of these sets is the new RS, as shown in Figure 1. To ensure convergence to x_0, we introduce a positive definite function U with $U(0) = 0$. For every $i > 0$, we require that $\Omega_1^i \subseteq \Omega_2^j$ and $U(j) < U(i)$ for some $j \neq i$, so that a trajectory starting in Ω_2^i eventually reaches Ω_2^j. We compute the control using the RCLF indexed by

$$i^*(x) \doteq \arg \min_{\{j \mid x \in \Omega_2^j\}} U(j). \tag{4}$$

Algorithm 1 We propose the following complete solution procedure for Problem 1.

I. Choose trim points and design RCLFs. On each iteration $i \geq 0$, do the following.

   A. If $i > 0$, find a trim point x_i such that $x_i \in \text{int}(\Omega_2^j)$ for some $j < i$.

   B. Select a quadratic RCLF $V_i(x)$ and stability margin $W_i(x)$ based on the linearized dynamics about x_i and a local optimization problem.

   C. Find $\Omega_1^i \subset \Omega_2^i$ such that the *nonlinear* system is RUAS(Ω_2^i,Ω_1^i). If $i > 0$, we also require $\Omega_1^i \subseteq \Omega_2^j$; otherwise, go back to step I.A using a new trim point.

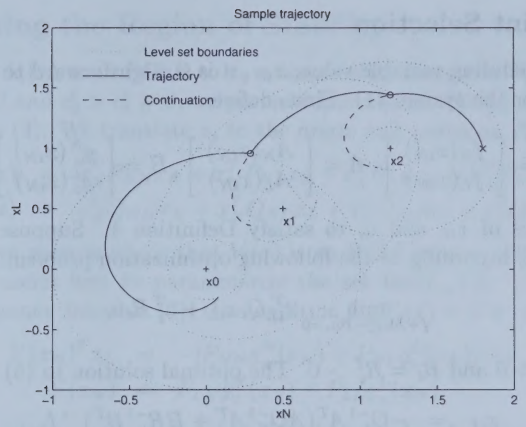


Figure 1: Sample trajectory to illustrate switching between level sets. The solid curve is a trajectory beginning at the point marked by an X; the circles are switching points. The dashed curves are continuations of the trajectory toward intermediate trim points.

D. If $i > 0$, set $U(i) = U(j) + 1$; otherwise, set $U(0) = 0$.

E. Repeat until the region $\mathcal{X} \doteq \cup_{j=0}^{i} \Omega_2^j$ covered by level sets is satisfactory.

II. Implement the control law. If the state is at a point $x \in \mathcal{X}$, do the following.

A. Select one of the level sets in which the point x lies.

B. Apply the inverse optimal control law [6] to get $u = \mu(x)$.

The following two results show that Algorithm 1 solves Problem 1 and that we can construct $\mathcal{X}$ to contain any compact trim surface if $\mathcal{W}$ is sufficiently small.

Theorem 1 Suppose there exist trim points $x_0, \ldots, x_N$ of the system (2); functions $V_0(x), \ldots, V_N(x)$, $W_0(x), \ldots, W_N(x)$; constants $c_2^0 > c_1^0 \geq 0, c_2^1 > c_1^1 > 0, \ldots, c_2^N > c_1^N > 0$; and a positive definite function $U : \{0, \ldots, N\} \to \mathbf{R}$; satisfying the following.

1. For each i, $V_i(x)$ is an RCLF to x_i with stability margin $W_i(x)$ over $V_i^{-1}[c_1^i, c_2^i]$.

2. For each $i \neq 0$, there exists $j \neq i$ such that $V_i^{-1}[0, c_1^i] \subseteq V_j^{-1}[0, c_2^j]$ and $U(j) < U(i)$.

Then a control law exists which makes the system RUAS($\mathcal{X}$,Ω) with $\mathcal{X} \doteq \cup_{i=0}^{N} V_i^{-1}[0, c_2^i]$ and $\Omega \doteq V_0^{-1}[0, c_1^0]$.

Theorem 2 If the linearized dynamics about each point in a compact trim surface $\mathcal{M}$ of the system (2) are stabilizable, then there exist $K > 0$; points $x_0, \ldots, x_N \in \mathcal{M}$; symmetric positive definite matrices $P_0, \ldots, P_N$, $Q_0, \ldots, Q_N$; constants $c_2^0 > c_1^0 \geq 0, \ldots, c_2^N > c_1^N \geq 0$; and a positive definite function $U : \{0, \ldots, N\} \to \mathbf{R}$; satisfying the following.

1. For each i, $V_i(x) \doteq (x - x_i)^T P_i (x - x_i)$ is an RCLF to x_i with stability margin $W_i(x) \doteq (x - x_i)^T Q_i (x - x_i)$ over $V_i^{-1}[c_1^i, c_2^i]$ when $\mathcal{W} = \{w \mid \|w\|_2 \leq K\}$.

2. $\mathcal{M} \subseteq \cup_{i=0}^{N} V_i^{-1}[0, c_2^i)$.

3. For each $i \neq 0$, there exists $j \neq i$ such that $V_i^{-1}[0, c_1^i] \subset V_j^{-1}[0, c_2^j)$ and $U(j) < U(i)$.

If $\mathcal{W} = \{0\}$, then we can arrange for this statement to hold with $c_1^i = 0$ for all i.

2.1 Trim Point Selection

Given a desired scheduling variable value, x_{iN}, it is straightforward to find a corresponding trim point x_i for the system (1). First, define

$$f \doteq \begin{bmatrix} f_N(x_{iN}) \\ f_L(x_{iN}) \end{bmatrix}, A \doteq \begin{bmatrix} A_N(x_{iN}) \\ A_L(x_{iN}) \end{bmatrix}, B \doteq \begin{bmatrix} g_u^N(x_{iN}) \\ g_u^L(x_{iN}) \end{bmatrix}.$$

We now seek values of x_{iL} and u_i to satisfy Definition 4. Suppose we would like to minimize x_{iL} and u_i according to the following optimization problem.

$$\min_{f+Ax_{iL}+Bu_i=0} x_{iL}^T Q_t x_{iL} + u_i^T R_t u_i \tag{5}$$

for some $Q_t = Q_t^T > 0$ and $R_t = R_t^T > 0$. The optimal solution to (5) is given by

$$\begin{aligned} x_{iL} &= -Q_t^{-1}A^T(AQ_t^{-1}A^T + BR_t^{-1}B^T)^{-1}f, \\ u_i &= -R_t^{-1}B^T(AQ_t^{-1}A^T + BR_t^{-1}B^T)^{-1}f, \end{aligned}$$

where $[A \mid B]$ has full row rank. The desired values of x_{iN} are found by iterating over grid points in the scheduling variable space to find, at each iteration i, a trim point such that $V_j(x_i) < c_2^j$ for some $j < i$. Detailed algorithms for this procedure appear in [11].

2.2 Local Control Lyapunov Functions

Next we construct an RCLF to x_i. The Lyapunov linearization theorem [19] yields a natural quadratic Lyapunov function based on the linearized dynamics of the system. Therefore, we propose a quadratic Lyapunov function $V_i(x) = (x - x_i)^T P_i (x - x_i)$ and stability margin $W_i(x) = (x - x_i)^T Q_i (x - x_i)$, where $P_i = P_i^T > 0$ and $Q_i = Q_i^T > 0$.

One way to select an RCLF and stability margin is to require the closed loop system locally to approximate the solution to an optimal control problem with some cost

$$J(x_0) = \int_0^\infty [(x - x_i)^T Q_c (x - x_i) + (u - u_i)^T R_c (u - u_i)]\, dt$$

based on $Q_c = Q_c^T > 0$ and $R_c = R_c^T > 0$ for the linearized system

$$\dot{x} = A(x - x_i) + B_w w + B_u(u - u_i).$$

Therefore, we find P_i to be the stabilizing solution to the algebraic Riccati equation.

$$A^T P_i + P_i A + Q_c - P_i B_u R_c^{-1} B_u^T P_i = 0. \tag{6}$$

For the closed loop system with the optimal control (LQR), the stability margin is given by $Q_i = Q_c + P_i B_u R_c^{-1} B_u^T P_i$ when $w \equiv 0$. In the nonlinear system, it may only be possible to achieve a stability margin $\alpha W_i(x)$, for some $\alpha \in (0, 1)$, over some set $V_i^{-1}[c_1^i, c_2^i]$.

2.3 Level Set Contained in Known Region of Stability

If $i > 0$, Theorem 1 requires that $V_i^{-1}[0, c_1^i] \subseteq V_j^{-1}[0, c_2^j]$ for some j. Given c_2^j, c_1^i is found by using the $\mathcal{S}$-procedure as discussed in [3] to solve the following optimization problem.

$$c_1^i \doteq \min_{V_j(x) \geq c_2^j} V_i(x).$$

2.4 Computing the Region of Stability

To check whether $V_i(x)$ is an RCLF to x_i with stability margin $\alpha W_i(x)$ over $V_i^{-1}[c_1^i, c_2^i]$ for given $\alpha \in (0,1]$ and $c_2^i > c_1^i \geq 0$, we parameterize the set $V_i^{-1}[c_1^i, c_2^i] \cap \ker(L_{g_u} V_i)$ and evaluate condition (3). We translate x_i to the origin and partition P_i and Q_i to obtain

$$\begin{aligned} V_i(x) &= x_N^T P_{NN} x_N + x_N^T P_{NL} x_L + x_L^T P_{LN} x_N + x_L^T P_{LL} x_L, \quad (7) \\ W_i(x) &= x_N^T Q_{NN} x_N + x_N^T Q_{NL} x_L + x_L^T Q_{LN} x_N + x_L^T Q_{LL} x_L. \quad (8) \end{aligned}$$

Given $\alpha \in (0,1]$, we want to check that $V_i(x)$ is an RCLF with stability margin $\alpha W_i(x)$. To do this, it is useful first to parameterize the set $\ker(L_{g_u} V_i)$. For a system of the form (1) and Lyapunov function (7), the condition $L_{g_u} V_i(x) = 0$ is equivalent to

$$\begin{aligned} Y(x_N)^T x_L &= -[P_{NN} g_u^N(x_N) + P_{NL} g_u^L(x_N)]^T x_N, \\ Y(x_N) &\doteq P_{LN} g_u^N(x_N) + P_{LL} g_u^L(x_N). \end{aligned}$$

To simplify the algebra, we assume that $Y(x_N)$ has rank m for all $x_N \in \mathbf{R}^k$. Theorem 3 shows how to parameterize $\ker(L_{g_u} V_i)$ by x_N and a parameter $\lambda \in \mathbf{R}^{n-k-m}$.

Theorem 3 Given a function $V_i(x)$ of the form (7),

$$\begin{aligned} \ker(L_{g_u} V_i) &= \left\{ \begin{bmatrix} x_N \\ G(x_N)\lambda - P_{LL}^{-1} P_{LN} x_N - \xi(x_N) \end{bmatrix}, x_N \in \mathbf{R}^k, \lambda \in \mathbf{R}^{n-k-m} \right\}, \\ \xi(x_N) &\doteq P_{LL}^{-1} Y(x_N)[Y(x_N)^T P_{LL}^{-1} Y(x_N)]^{-1} g_u^N(x_N)^T R x_N, \\ R &\doteq P_{NN} - P_{NL} P_{LL}^{-1} P_{LN}, \end{aligned}$$

where $G(x_N) \in \mathbf{R}^{(n-k)\times(n-k-m)}$ is any matrix of full rank such that $Y(x_N)^T G(x_N) \equiv 0$.

Now we can express $V_i(x)$ restricted to $\ker(L_{g_u} V_i)$ as a function of x_N and λ. To analyze stability over the level set $V_i^{-1}[c_1^i, c_2^i]$, we need the following definitions.

$$\begin{aligned} V_i(x_N, \lambda) &= [x_N^T R x_N + \xi(x_N)^T P_{LL} \xi(x_N)] + \lambda^T G(x_N)^T P_{LL} G(x_N) \lambda, \\ \mathcal{Y}_i(c_2^i) &\doteq \{x_N \in \mathbf{R}^k \mid V_i(x_N, 0) \leq c_2^i\}, \\ \mathcal{Z}_i(c_1^i, c_2^i, x_N) &\doteq \{\lambda \in \mathbf{R}^{n-k-m} \mid c_1^i \leq V_i(x_N, \lambda) \leq c_2^i\}, \\ \Gamma_i(c_1^i, c_2^i, x_N) &\doteq \max_{\lambda \in \mathcal{Z}_i(c_1^i, c_2^i, x_N)} \sup_{w \in \mathcal{W}} L_f V_i(x) + L_{g_w} V_i(x) w + \alpha W_i(x). \end{aligned}$$

Proposition 1 $V_i(x)$ is an RCLF to the origin with stability margin $\alpha W_i(x)$ over the set $V_i^{-1}[c_1^i, c_2^i]$ if and only if $\Gamma_i(c_1^i, c_2^i, x_N) \leq 0$ for all $x_N \in \mathcal{Y}_i(c_2^i)$.

We check the condition in Proposition 1 by gridding $\mathcal{Y}_i(c_2^i)$ and checking $\Gamma_i(c_1^i, c_2^i, x_N) \leq 0$ for all x_N in the grid. For the system (1), we develop a procedure for solving this problem which requires substantially less computation time than the related problem of determining the RS for a general nonlinear system. This is made precise in Section 3.

We apply Theorem 3 to get the following.

$$\Gamma_i(c_1^i, c_2^i, x_N) = \max_{c_1^i \leq V_i(x_N, \lambda) \leq c_2^i} \sup_{w \in \mathcal{W}} a_0(x_N) + b_0(x_N)^T \lambda + \lambda^T C_0(x_N) \lambda + w^T [s(x_N) + T(x_N)\lambda].$$

The coefficients can be found by straightforward algebra. If $\mathcal{W} = \{0\}$, we obtain

$$\Gamma_i(c_1^i, c_2^i, x_N) = \max_{c_1^i \leq V_i(x_N, \lambda) \leq c_2^i} a_0(x_N) + b_0(x_N)^T \lambda + \lambda^T C_0(x_N) \lambda. \quad (9)$$

This quadratic optimization problem with quadratic constraints can be solved using the $\mathcal{S}$-procedure as discussed in [3]. Theorem 4 shows that the two constraints are never simultaneously active; therefore, the $\mathcal{S}$-procedure is nonconservative [3].

Theorem 4 In the problem (9) with $x_N \in \mathcal{Y}_i(c_2^i)$, if $C_0(x_N) \not< 0$, then $\Gamma_i(c_1^i, c_2^i, x_N) = \Gamma_i(0, c_2^i, x_N)$. If $C_0(x_N) < 0$, one of the following holds for $\lambda^* = -\frac{1}{2}C_0(x_N)^{-1}b_0(x_N)$.

1. If $V_i(x_N, \lambda^*) < c_1^i$, then $\Gamma_i(c_1^i, c_2^i, x_N) = \Gamma_i(c_1^i, \infty, x_N)$.
2. If $V_i(x_N, \lambda^*) > c_2^i$, then $\Gamma_i(c_1^i, c_2^i, x_N) = \Gamma_i(0, c_2^i, x_N)$.
3. If $c_1^i \leq V_i(x_N, \lambda^*) \leq c_2^i$, then $\Gamma_i(c_1^i, c_2^i, x_N) = a_0(x_N) - \frac{1}{4}b_0(x_N)^T C_0(x_N)^{-1}b_0(x_N)$.

If $\mathcal{W}$ is a convex polytope, robust stability can be analyzed using Theorem 4 with each of the extreme points of $\mathcal{W}$ substituted for w because the condition is affine in w.

2.5 Iteration over Level Values

Since $\Gamma_i(c_1^i, c_2^i, x_N)$ is nondecreasing in c_2^i, a bisection can be used to find the largest c_2^i satisfying $\Gamma_i(c_1^i, c_2^i, x_N) \leq 0$ over $x_N \in \mathcal{Y}_i(c_2^i)$. If this condition is violated for all $c_2^i > c_1^i$, either a different trim point must be used, or the set $\mathcal{W}$ is too large. At x_0, we need both c_1^0 and c_2^0 so that $\Gamma_0(c_1^0, c_2^0, x_N) \leq 0$ over $x_N \in \mathcal{Y}_0(c_2^0)$. If $\mathcal{W} = \{0\}$, then $c_1^0 = 0$, and we find c_2^0 by bisection. Otherwise, a procedure such as the one outlined in [11] is required.

2.6 Control Law Implementation

To implement the control law, we first need the index i with the smallest value of $U(i)$ such that $V_i(x) \leq c_2^i$. The index $i^*(x)$ given by (4) gives us the correct RCLF to use in the control input computation. The largest stability margin achieved over a given level set depends on the corresponding level value, so the stability margin should be scaled by a continuous positive function $\alpha(V_i(x))$. In this way, the control can achieve the maximum guaranteed stability margin over any level set up to the RS. While the optimal $\alpha^*(V_i(x))$ is hard to determine, we can compute level values for a finite number of α, and then interpolate in such a way that $\alpha \leq \alpha^*$, with $\alpha = \alpha^*$ at the interpolation points. As the partition of $\alpha \in (0, 1]$ is refined, the optimal stability margin is recovered asymptotically.

3 Computational Complexity

The computation times for each step in Algorithm 1 are polynomial in n for fixed k. This is a significant savings over the related problem of gridding the state space to find an RS for a general nonlinear system, which is exponential in n. Trim point selection involves gridding over x_N and solving the optimization problem (5) at each grid point. If there are N_g grid points for each dimension in x_N, the computation time is approximately $N_g^k \mathcal{O}(n^4)$. The time required to solve the Riccati equation (6) for $V_i(x)$ and $W_i(x)$ is $\mathcal{O}(n^2)$. To analyze stability over a given level set for the system (1), we grid $x_N \in \mathcal{Y}_i(c_2^i)$ and solve an LMI problem to compute $\Gamma_i(c_1^i, c_2^i, x_N)$ at each grid point. In the case $\|w\|_\infty \leq 1$, the computation time to solve this problem using an ellipsoid algorithm [3] is roughly $N_c^k 2^l \mathcal{O}(n^3)$, where N_c is the number of grid points in $\mathcal{Y}_i(c_2^i)$ for each dimension.

4 Cart-pole Example

Consider the system shown in Figure 2. A pole is hinged on a cart, and a spring joins the top of the pole to a fixed point on the wall behind the cart. The control is a force on

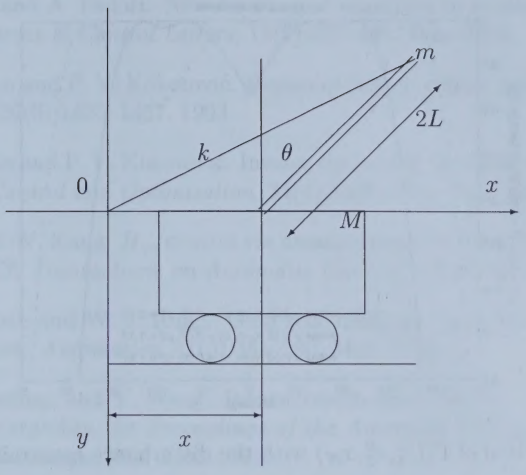


Figure 2: Cart with inverted pendulum and spring.

the cart and the disturbance is a torque on the pole. We want a control design to drive the system to the origin from an initial condition. A simplified model is given by

$$\begin{bmatrix} \dot{\theta} \\ \dot{q} \\ \dot{x} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{mL\sin\theta([M+m]g+2kL\cos\theta)}{I_z(M+m)-m^2L^2\cos^2\theta} \\ 0 \\ \frac{-L\sin\theta(m^2gL\cos\theta+2kI_z)}{I_z(M+m)-m^2L^2\cos^2\theta} \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & \frac{-kL\cos\theta(2M+m)}{I_z(M+m)-m^2L^2\cos^2\theta} & 0 \\ 0 & 0 & 1 \\ 0 & \frac{k(2mL^2\cos^2\theta-I_z)}{I_z(M+m)-m^2L^2\cos^2\theta} & 0 \end{bmatrix} \begin{bmatrix} q \\ x \\ v \end{bmatrix} +$$
$$\begin{bmatrix} 0 \\ \frac{M+m}{I_z(M+m)-m^2L^2\cos^2\theta} \\ 0 \\ \frac{-mL\cos\theta}{I_z(M+m)-m^2L^2\cos^2\theta} \end{bmatrix} w + \begin{bmatrix} 0 \\ \frac{-mL\cos\theta}{I_z(M+m)-m^2L^2\cos^2\theta} \\ 0 \\ \frac{I_z}{I_z(M+m)-m^2L^2\cos^2\theta} \end{bmatrix} u.$$

After designing RCLF $V_0(x)$ and stability margin $W_0(x)$ for the linearized dynamics about $x_0 = 0$ as in Section 2.2, we analyze robust stability with disturbances $|w| \leq w_{max}$ and a stability margin scaling $\alpha = 0.1$. The results are plotted in Figure 3 for several values of w_{max}. We can repeat this to get $c_1^0(\alpha)$ and $c_2^0(\alpha)$ over a range of α. We expand the RS using RCLFs to multiple trim points, as shown in Figure 4 for $\mathcal{W} = \{0\}$.

5 Conclusions

Progress on the nonlinear control problem is difficult due to the inherent complexity of the class of arbitrary nonlinear systems. The new method is a stabilizing alternative to gain scheduling for the class of systems (1), which is sufficiently restricted so that computations can be made tractable. The computation time is polynomial in the state dimension, and the control can be designed for robustness to bounded disturbances.

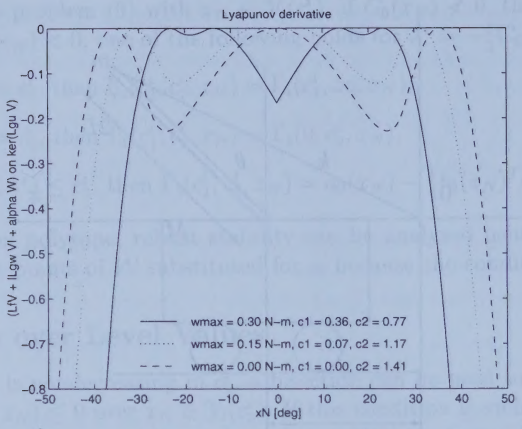

Lyapunov derivative
(LfV + ||Lgw Vl + alpha W) on ker(Lgu V)
xN [deg]
wmax = 0.30 N–m, c1 = 0.36, c2 = 0.77
wmax = 0.15 N–m, c1 = 0.07, c2 = 1.17
wmax = 0.00 N–m, c1 = 0.00, c2 = 1.41

Figure 3: Variation of $\Gamma_0(c_1^0, c_2^0, x_N)$ with the disturbance constraint $|w| \leq w_{max}$.

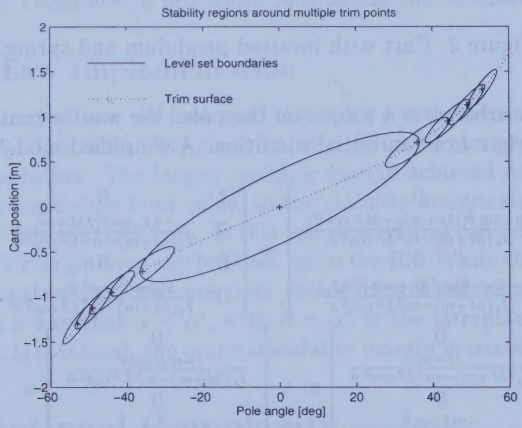


Figure 4: Regions of stability about multiple trim points for the cart-pole.

References

[1] J. A. Ball and A. J. van der Schaft. *J*-inner-outer factorization, *J*-spectral factorization, and robust control for nonlinear systems. *IEEE Transactions on Automatic Control*, 41(3):379–392, Mar. 1996.

[2] S. Battilotti. Global output regulation and disturbance attenuation with global stability via measurement feedback for a class of nonlinear systems. *IEEE Transactions on Automatic Control*, 41(3):315–327, Mar. 1996.

[3] S. Boyd, L. El Ghaoui, E. Feron, and V. Balakrishnan. *Linear Matrix Inequalities in System and Control Theory.* Society for Industrial and Applied Mathematics, Philadelphia, PA, 1994.

[4] C. I. Byrnes and A. Isidori. New results and examples in nonlinear feedback stabilization. *Systems & Control Letters*, 12(5):437–442, June 1989.

[5] R. A. Freeman and P. V. Kokotović. Design of 'softer' robust nonlinear control laws. *Automatica*, 29(6):1425–1437, 1993.

[6] R. A. Freeman and P. V. Kokotović. Inverse optimality in robust stabilization. *SIAM Journal on Control and Optimization*, 34(4):1365–1391, July 1996.

[7] A. Isidori and W. Kang. H_∞ control via measurement feedback for general nonlinear systems. *IEEE Transactions on Automatic Control*, 40(3):466–472, Mar. 1995.

[8] D. A. Lawrence and W. J. Rugh. Gain scheduling dynamic linear controllers for a nonlinear plant. *Automatica*, 31(3):381–390, Mar. 1995.

[9] Y. Lin, E. Sontag, and Y. Wang. Recent results on Lyapunov-theoretic techniques for nonlinear stability. In *Proceedings of the American Control Conference*, pages 1771–1775, Baltimore, MD, June 1994.

[10] R. Marino and P. Tomei. Robust stabilization of feedback linearizable time-varying uncertain nonlinear systems. *Automatica*, 29(1):181–189, 1993.

[11] M. W. McConley, B. D. Appleby, M. A. Dahleh, and E. Feron. A control Lyapunov function approach to robust stabilization of nonlinear systems. Submitted to *IEEE Transactions on Automatic Control*, 1996.

[12] L. Praly, B. d'Andréa Novel, and J.-M. Coron. Lyapunov design of stabilizing controllers for cascaded systems. *IEEE Transactions on Automatic Control*, 36(10):1177–1181, Oct. 1991.

[13] W. J. Rugh. Analytical framework for gain scheduling. *IEEE Control Systems Magazine*, 11(1):79–84, Jan. 1991.

[14] J. S. Shamma and M. Athans. Guaranteed properties of gain scheduled control for linear parameter-varying plants. *Automatica*, 27(3):559–564, May 1991.

[15] E. D. Sontag. A 'universal' construction of Artstein's theorem on nonlinear stabilization. *Systems & Control Letters*, 13(2):117–123, Aug. 1989.

[16] J. Tsinias. A theorem on global stabilization of nonlinear systems by linear feedback. *Systems & Control Letters*, 17(5):357–362, Nov. 1991.

[17] J. Tsinias. Versions of Sontag's "input to state stability condition" and the global stabilizability problem. *SIAM Journal on Control and Optimization*, 31(4):928–941, July 1993.

[18] A. J. van der Schaft. L_2-gain analysis of nonlinear systems and nonlinear state feedback H_∞ control. *IEEE Transactions on Automatic Control*, 37(6):770–784, June 1992.

[19] M. Vidyasagar. *Nonlinear Systems Analysis*. Prentice-Hall, Inc., Englewood Cliffs, NJ, second edition, 1993.

Blind Error-Bound Estimation for OBE Identification

M. Nayeri, T. M. Lin, and J. R. Deller, Jr.
Department of Electrical Engineering / 260 EB
Michigan State University
E. Lansing, Michigan 48824-1226 USA

Abstract. Conventional *optimal bounding ellipsoid* (OBE) algorithms require an *a priori* description of error bounds that is often unknown. Conservative (over-estimated) error bounds used in practice result in theoretically inconsistent parameter estimation. This paper presents an enhanced OBE algorithm that is consistently convergent without *a priori* knowledge of error bounds. Only a lower bound of the "tail probability" of the disturbance process is required. Conventional and enhanced OBE algorithms are compared in simulations.

1 Introduction

Bounded-error estimation refers to a broad class of parameter-estimation techniques in which the modeling or measurement errors are assumed bounded but otherwise unknown. In this paper, we focus on a special class of bounded-error methods, the OBE algorithms. The OBE approach is of particular interest in real-time signal-processing and control applications, and is appealing in its *formal* similarity to the widely-used *recursive least squares* (RLS) algorithm (e.g., [10]) and its variants.

With respect to RLS-like methods, OBE algorithms have been shown to have superior tracking in non-stationary dynamics, and to provide more accurate and quickly converging solutions in the stationary, limited data case (e.g., [2]). Further, the OBE algorithms can be structured to be more computationally efficient than RLS and similar methods [2]. Additionally, OBE algorithms yield a feasible *set* of solutions, whereas the centroid of the set (computed recursively) provides a point estimate that is interpretable as a weighted least-square-error estimate.

The benefits of OBE algorithms notwithstanding, a significant impediment to their practical application is the need for very accurate model error bounds. While such bounds occur naturally in some problems, they are difficult to ascertain in many others in which the OBE approach would be desirable. An example occurs in the determination of bounds for parametric coding of speech signals in which there is no inherent means of determining proper bounds based on physical considerations [4],[7, Ch. 29]. Failure to prescribe appropriate bounds is potentially catastrophic. Deterministically, underestimated bounds may cause the estimate to diverge to a parameter vector that is outside the *true* feasible set, while overestimated bounds may cause the estimator to "freeze" at a value far away from the true parameters. In either case, improper bounds lead to a statistically inconsistent estimator.

This paper presents an enhanced OBE algorithm with an *automatic bound estimation* (ABE) procedure. The new "OBE-ABE" algorithm obviates the need for accurate

error bounds, thus removing the severe practical limitation described above. The ABE enhancement may be incorporated into *any* OBE algorithm [3],[7, Ch. 4]. The *set-membership stochastic approximation* (SM-SA) OBE algorithm [3] is used here because this formulation is amenable to the proof of key results.

2 Formulation and Background

Many systems can be modeled by a *linear-in-parameters* model

$$y_n = \theta_*^T \mathbf{x}_n + v_n \tag{1}$$

in which $\theta_* = [a_1, \cdots, a_p, b_0, b_1, \cdots, b_q]$ is the "true" parameter vector to be identified; $\{\mathbf{x}_n\}$ is a sequence of measurable vectors of dimension $m = p + q + 1$; and $\{v_n\}$ is an error sequence, ordinarily assumed to be stationary and uncorrelated. An important special case is the *auto-regressive with exogenous input* (ARX) model [10] in which $\mathbf{x}_n = [y_{n-1}, \cdots, y_{n-p}, u_n, \cdots, u_{n-q}]$ is the observed data set composed of values of the input sequence $\{u_n\}$ and output $\{y_n\}$. The present work applies to the ARX model with $\{v_n\}$ composed of bounded, independent, and stationary (hence, independent identically distributed, i.i.d.) random variables. The results of this paper are generalizable to cases in which y_n and v_n are vectors [2].

OBE algorithms assume that $\{v_n\}$ has known pointwise energy bounds,

$$v_n^2 \leq \gamma_n, \quad \forall n \in \mathbf{N}. \tag{2}$$

Given (2), it can be demonstrated that $\theta_* \in \Omega_n$ where Ω_n is a hyper-ellipsoidal set,

$$\Omega_n \stackrel{\text{def}}{=} \left\{\theta \middle| (\theta - \theta_n)^T \mathbf{P}_n^{-1} (\theta - \theta_n) < \kappa_n\right\} \tag{3}$$

where the ellipsoid center, θ_n, and the defining matrix $\mathbf{P}_n$ are computed recursively using

$$G_n = \mathbf{x}_n{}^T \mathbf{P}_{n-1} \mathbf{x}_n \tag{4}$$

$$\varepsilon_n = y_n - \theta_{n-1}^T \mathbf{x}_n \tag{5}$$

$$\mathbf{P}_n = \frac{1}{\alpha_n}\left[\mathbf{P}_{n-1} - \frac{\beta_n \mathbf{P}_{n-1} \mathbf{x}_n \mathbf{x}_n{}^T \mathbf{P}_{n-1}}{\alpha_n + \beta_n G_n}\right] \tag{6}$$

$$\theta_n = \theta_{n-1} + \beta_n \mathbf{P}_n \mathbf{x}_n \varepsilon_n \tag{7}$$

$$\kappa_n = \alpha_n \kappa_{n-1} + \beta_n \gamma_n - \frac{\alpha_n \beta_n \varepsilon_n^2}{\alpha_n + \beta_n G_n}. \tag{8}$$

Recursions (4) – (8) comprise a general OBE algorithm. The ellipsoid center θ_n is an estimator of the parameters θ_* at each n. The process is initialized with $\theta_0 = \mathbf{0}$, $\kappa_0 = \mu$ and $\mathbf{P}_0 = \frac{1}{\mu^2}\mathbf{I}$, where μ is a small number, typically 10^{-3}. $\{\alpha_n\}$ and $\{\beta_n\}$ are positive weighting sequences that differ across OBE algorithms. Excepting the OBE algorithm described in [1], the weights are chosen to minimize the "size" of the set Ω_n at each iteration. When such optimal weights do not exist, the updating need not take place. This is a critical feature of OBE algorithms.

In SM-SA, the volume of the ellipsoid (proportional to $\det \kappa_n \mathbf{P}_n$) is minimized at each iteration by letting $\beta_n = \lambda_n$, $\alpha = 1 - \lambda_n$, and seeking the optimal λ_n, given by

$$\lambda_n = \left\{\frac{-b_n + \sqrt{b_n^2 - 4a_n c_n}}{2a_n}, \quad \text{if} \quad c_n < 0; \quad 0 \quad \text{if} \quad c_n \geq 0\right\}, \tag{9}$$

in which

$$\begin{aligned} a_n &= m\gamma_n - m\varepsilon_n^2 + mG_n^2\gamma_n - 2mG_n\gamma_n - \kappa_{n-1}G_n + \kappa_{n-1}G_n^2 + G_n\gamma_n - G_n^2\gamma_n - \varepsilon_n^2 G_n \\ b_n &= 2m\varepsilon_n^2 - 2m\gamma_n + 2mG_n\gamma_n + 2\kappa_{n-1}G_n - \kappa_{n-1}G_n^2 - G_n\gamma_n + \varepsilon_n^2 G_n \\ c_n &= m\gamma_n - m\varepsilon_n^2 - \kappa_{n-1}G_n. \end{aligned} \tag{10}$$

Choosing a proper bounding sequence $\{\gamma_n\}$ is critical in practice. Bounds underestimated at one or more n violate (2), and render the optimization process invalid. In this case a statistical analysis of the estimator is not possible. Violation of (2) may cause the true parameter vector to be permanently removed from the set Ω_n with unpredictable consequences for the estimate (see [2]). Practitioners therefore attempt to be conservative in applying bounds, "erring" on the side of over-bounding.

A conservative sequence of bounds $\{\gamma_n\}$ will assure a meaningful feasibility set Ω_n at each n in the sense that $\theta_* \in \Omega_n$. However, recent work of Liu [6] and Nayeri *et al.* [8] shows that achieving an ultimately small set Ω_n depends explicitly on a set of very tight bounds. This means that the estimator may be very imprecise, even asymptotically, if the bounds are too "loose." Formally, it has been shown that if a *persistency of excitation* (PE) [10] condition holds, then the estimator of any OBE algorithm converges to a finite neighborhood of the true parameter vector θ_* [8]. However, consistency of the estimator (reduction of Ω_n to a point) can be proved only when exact bounds are employed for infinitely many samples. A formal statement, proved in [6] and [8], is as follows:

Lemma 1 *Assume that the PE condition holds i.e., there exists an* $N_1 \in \mathsf{N}$ *and* $\rho_1, \rho_2 > 0$ *such that, for all* n, $\rho_1 \mathbf{I} \leq \sum_{k=n+1}^{n+N_1} \mathbf{x}_k \mathbf{x}_k{}^T \leq \rho_2 \mathbf{I}$. *If the bounds of (2) are over-estimated, i.e., if there exists an* $\epsilon > 0$, *such that* $\gamma_n - v_n{}^2 > \epsilon$, $\forall n \in \mathsf{N}$, *then the ellipsoid does not shrink to a point asymptotically.* □

According to Lemma 1, the OBE estimator is not guaranteed to be consistent when conservative bounds are used. The OBE-ABE algorithm presented below relaxes the need for precise *a priori* bounds in order to obtain a consistent estimator.

3 The OBE-ABE Algorithm

The OBE-ABE algorithm applies to the ARX model of form (1) with an i.i.d. sequence $\{v_n\}$ whose true (but unknown) amplitude bounds are symmetrical (also constant because of the i.i.d. assumption): $-\sqrt{\gamma_*} \leq v_n \leq \sqrt{\gamma_*}$. The probability space on which all the stochastic quantities in this paper are based is $(\mathsf{R}, \mathcal{B}, P)$ where R is the set of real numbers, $\mathcal{B}$ the Borel set, and P a probability measure.

The following lemmas are used in proving the consistent convergence of OBE-ABE. The first is a key result due to Veres and Norton [11], and Nayeri *et al.* [6, 8] which asserts that if the error sequence visits arbitrarily small neighborhoods of the bounds with nonzero probability, then the OBE estimator is consistent. Critically, the convergence does not depend on initiating the algorithm with the true bounds.

Lemma 2 (Proof: [6]). *Assume that the* γ_n *are set to a known* γ_* *in the OBE estimation process. Given* $\epsilon > 0$, *let* $D_{\epsilon+} = [\sqrt{\gamma_*} - \epsilon, \sqrt{\gamma_*}]$ *and* $D_{\epsilon-} = [-\sqrt{\gamma_*}, -\sqrt{\gamma_*} + \epsilon]$. *If the PE condition of Lemma 1 holds and, if there exists a* $\delta > 0$ *such that*

$$P(v_n \in D_{\epsilon+}) > \delta, \text{ and } P(v_n \in D_{\epsilon-}) > \delta \quad \forall n, \tag{11}$$

then the estimator, θ_n, of any volume-optimizing OBE algorithm converges consistently to the true parameter vector, θ_*. Further, consistent convergence is obtained even if the bounds are overestimated (i.e., $\gamma_n > \gamma_*$, for all n) as long as $\gamma_n \to \gamma_*$ as $n \to \infty$. □

The following lemma analytically portrays the fact that the OBE algorithm operating on a persistently-excited system will ultimately cease to find innovation in the data (hence $\lambda_n = 0$ in SM-SA). The negativity of c_n of (10) is the indicator for innovation (e.g., [3]).

Lemma 3 (Proof: [6]). *Let $c_n^- = \{c_n$, if $c_n < 0$; 0, otherwise$\}$, and $c_n^+ = \{c_n$, if $c_n > 0$; 0, otherwise$\}$. If the PE condition of Lemma 1 holds, then $\lim_{n\to\infty} c_n^- = 0$.* □

The next lemma asserts that if the error bounds are overestimated, then the time between updates becomes infinite as $n \to \infty$.

Lemma 4 (Proof: [5]) *Assume that the PE condition of Lemma 1 holds. If the bounds γ_n of the OBE algorithm are overestimated, i.e., if there exists an $\epsilon_1 > 0$ such that for all n, $\gamma_n - {v_n}^2 > \epsilon_1$, then the expected time between updates of the algorithm approaches infinity as $n \to \infty$.* □

Since an OBE algorithm updates its estimator only when $c_n < 0$, the probability of an update diminishes as $n \to \infty$ [5]. This implies that the expected updating interval approaches infinity as time increases indefinitely.

The OBE-ABE algorithm appearing in Table 1 is justified inherently in the proof of the following theorem.

I. Initialization:
   1. $\theta_0 = \mathbf{0}$, $\kappa_0 = \mu$ and $\mathbf{P}_0 = \frac{1}{\mu^2}\mathbf{I}$, where μ is a small number, typically 10^{-3}.
   2. $\gamma_0 =$ any overestimated bound.
   3. Choose ϵ (small positive number), N and M (as in Remark 2 following Theorem 1).

II. Recursion:
   For $n = 1 : M$
      If $c_n < 0$
         Execute equations (4) - (8).
      Otherwise,
         If a time interval $\mathcal{I}$ of length N over which $c_n \geq 0$ is found,

$$\gamma_n = \{\gamma_{n-1} - d_J \text{ if } d_J > 0; \quad \gamma_{n-1} \text{ otherwise,}\}$$

where $d_J \stackrel{\text{def}}{=} \kappa_{J-1} G_J / m - \epsilon(2\sqrt{\gamma_{n-1}} - \epsilon)$ and $J = \arg\max_{n\in\mathcal{I}} \varepsilon_n^2$.

Table 1: The OBE with automatic bound estimation (OBE-ABE) algorithm.

Theorem 1 *Assume that v_n in system (1) is i.i.d. with* unknown *bounds $\sqrt{\gamma_*}$ and $-\sqrt{\gamma_*}$. If the PE condition of Lemma 1 holds and, if given $\epsilon > 0$, there exists a $\delta > 0$ such that probability inequalities (11) hold, then the estimator of the OBE-ABE algorithm (Table 1) converges consistently.*

Proof: Let $\mathcal{I}$ be a time interval of length $N \in \mathbf{N}$ over which the OBE algorithm does not update its estimator. By (10),

$$c_n = m(\gamma_n - \varepsilon_n^2) - \kappa_{n-1} G_n \geq 0, \qquad \forall n \in \mathcal{I}. \tag{12}$$

Also from (5),

$$\varepsilon_n^2 = (y_n - \theta_{n-1}^T \mathbf{x}_n)^2 = (v_n - \tilde{\theta}_{n-1}^T \mathbf{x}_n)^2 \tag{13}$$

where $\tilde{\theta}_{n-1} = \theta_{n-1} - \theta_*$.

Let A_n^- be the event that $v_n \in D_{\epsilon^-}$ and $\tilde{\theta}_{n-1}^T \mathbf{x}_n \geq 0$. Let A_n^+ be the event that $v_n \in D_{\epsilon^+}$ and $\tilde{\theta}_{n-1}^T \mathbf{x}_n < 0$. Then, $A_n \stackrel{\text{def}}{=} A_n^+ \cup A_n^-$ is the event that the distance between v_n and $\tilde{\theta}_{n-1}^T \mathbf{x}_n$ is greater than or equal to $\sqrt{\gamma_*} - \epsilon$. Denote by A_n^c the complement of A_n. v_n and $\mathbf{x}_n$ are independent since v_n is i.i.d. Hence, v_n and $\tilde{\theta}_{n-1}^T \mathbf{x}_n$ are independent. Therefore, $\{v_n \in D_{\epsilon^-}\}$ (or $\{v_n \in D_{\epsilon^+}\}$) and A_{n-1}^c are independent.

Let $P(v_n \in D_{\epsilon^-}) = p^-$ ($> \delta$ by assumption), and $P(v_n \in D_{\epsilon^+}) = p^+$ ($> \delta$). Then, $p \stackrel{\text{def}}{=} \min(p^+, p^-) > \delta$. Since A_n^+ and A_n^- are mutually exclusive for sufficiently small ϵ,

$$\begin{aligned} P(A_n) &= P(A_n^-) + P(A_n^+) = P(v_n \in D_{\epsilon^-}) P(\tilde{\theta}_{n-1}^T \mathbf{x}_n \geq 0) \\ &\qquad + P(v_n \in D_{\epsilon^+}) P(\tilde{\theta}_{n-1}^T \mathbf{x}_n < 0) \\ &\geq p[\, P(\tilde{\theta}_{n-1}^T \mathbf{x}_n \geq 0) + 1 - P(\tilde{\theta}_{n-1}^T \mathbf{x}_n \geq 0)\,] = p > \delta. \end{aligned}$$

So, $P(A_n^c) = 1 - P(A_n) < 1 - \delta$. Also,

$$\begin{aligned} P(A_n^c | A_{n-1}^c) &= 1 - P(A_n | A_{n-1}^c) = 1 - [P(A_n^- | A_{n-1}^c) + P(A_n^+ | A_{n-1}^c)] \\ &= 1 - \{P(v_n \in D_{\epsilon^-}) P(\tilde{\theta}_{n-1}^T \mathbf{x}_n \geq 0 | A_{n-1}^c) \\ &\qquad + P(v_n \in D_{\epsilon^+}) P(\tilde{\theta}_{n-1}^T \mathbf{x}_n < 0 | A_{n-1}^c)\} \\ &\leq 1 - \{p[P(\tilde{\theta}_{n-1}^T \mathbf{x}_n \geq 0 | A_{n-1}^c) + 1 - P(\tilde{\theta}_{n-1}^T \mathbf{x}_n \geq 0) | A_{n-1}^c]\} \\ &= 1 - p < 1 - \delta. \end{aligned}$$

Since $\{v_n \in D_{\epsilon^-}\}$ (or $\{v_n \in D_{\epsilon^+}\}$) and A_{n-1}^c are independent,

$$P(A_n^c \cap A_{n-1}^c) = P(A_{n-1}^c) P(A_n^c | A_{n-1}^c) < (1 - \delta)^2.$$

Similarly, $P(A_n^c | A_{n-1}^c A_{n-2}^c) < 1 - \delta$. So,

$$P(A_n^c \cap A_{n-1}^c \cap A_{n-2}^c) = P(A_{n-1}^c A_{n-2}^c) P(A_n^c | A_{n-1}^c A_{n-2}^c) < (1 - \delta)^2 (1 - \delta) = (1 - \delta)^3.$$

By induction, $P(\cap_{i=1}^N A_i^c) < (1 - \delta)^N$. Hence,

$$P(A_n \text{ occurs at least once in } \mathcal{I}) = P(\cup_{i=1}^N A_i) = 1 - P(\cap_{i=1}^N A_i^c) > 1 - (1 - \delta)^N. \tag{14}$$

Note that the occurence of A_n at least once in $\mathcal{I}$ implies that $\max_{n \in \mathcal{I}} \varepsilon_n^2 \geq (\sqrt{\gamma_*} - \epsilon)^2$. Hence, from (13) and (14),

$$\max_{n \in \mathcal{I}} \varepsilon_n^2 \geq (\sqrt{\gamma_*} - \epsilon)^2 = \gamma_* - \epsilon(2\sqrt{\gamma_*} - \epsilon) \geq \gamma_* - \epsilon(2\sqrt{\gamma_{n-1}} - \epsilon) \tag{15}$$

with probability $1 - (1 - \delta)^N$ (> 0.99995 if $\delta = 0.01$ and $N > 1000$).

Let $J = \arg\max_{n \in \mathcal{I}} \varepsilon_n^2$. Then, by letting $n = J$, (12) can be written

$$\gamma_J - \varepsilon_J^2 - \frac{\kappa_{J-1} G_J}{m} \geq 0. \tag{16}$$

This result leads immediately to the recursion for γ_n in Table 1 and verifies that this value is an upper bound of v_n with probability $1-(1-\delta)^N$.

Let M be the total number of available observations. Note that the maximum number of updates of γ_n is less than M/N. We claim that, by letting $\epsilon \to 0$ and $M \to \infty$, $\kappa_n \to 0$ as $n \to \infty$ ($M \to \infty$) with probability greater than $[1-(1-\delta)^N]^{\frac{M}{N}}$. This can be proved as follows. If γ_n is overestimated, then, a time interval $\mathcal{I}$ of length N exists regardless of the size of n [5]. Therefore, γ_n would continue to be updated until $d_J \to 0$. This implies $\kappa_J \to 0$ since $G_n > 0$ due to the PE condition and stability assumption. It follows that $\kappa_n \to 0$ due to its non-increasing property.

Now, let $N = \sqrt{M}$. Then, the probability $[1-(1-\delta)^N]^{\frac{M}{N}} = [1-(1-\delta)^{\sqrt{M}}]^{\sqrt{M}} \to 1$ as $M \to \infty$. [Use L'Hopital's rule to show that $(1-q^n)^n \to 1$, as $n \to \infty$, for $q \in (0,1)$]. This proves consistent convergence of the OBE-ABE algorithm. □

Remark 1: Instead of exact knowledge of γ_*, the OBE-ABE algorithm requires only *a lower bound* (δ) of the "tail probability" of v_n [see ((11))] for consistency. Hence, the *a priori* condition of the OBE-ABE algorithm is less restrictive than that of OBE.

Remark 2: Since M is finite in practice, ϵ is first chosen to be a small positive number (κ_M decreases with ϵ in the recursions in Table 1). Given ϵ, a δ is obtained. Then, a large enough M and $N(=\sqrt{M}$, for example) are chosen such that the probability $[1-(1-\delta)^N]^{\frac{M}{N}}$ is close to one. For example, if $\delta = 0.1$, choose $N = 100$ and $M = 10,000$, for example, to get the probability $[1-(1-\delta)^N]^{\frac{M}{N}} > 0.997$. If $\delta = 0.02$, choose $N = 600$ and $M = 360,000$, for example, to get the probability $[1-(1-\delta)^N]^{\frac{M}{N}} > 0.996$. Since the probability $[1-(1-\delta)^N]^{\frac{M}{N}}$ is the most conservative value, if M is fixed in practice and is not too large ($M < 10,000$, for example), N can be chosen large enough so that $1-(1-\delta)^N$ is close to one.

The following corollary asserts the consistent convergence of OBE-ABE in cases in which the distribution of v_n instead of the bound γ_* is known.

Corollary 1 *Assume that v_n is i.i.d. with unknown bounds $\sqrt{\gamma_*}$ and $-\sqrt{\gamma_*}$. If the PE condition of Lemma 1 holds and if v_n is uniformly distributed, or if v_n is binary Bernoulli distributed with $P(v_n = \sqrt{\gamma_*}) > \delta$ and $P(v_n = -\sqrt{\gamma_*}) > \delta$, $\forall n$, then the parameter estimator of the OBE-ABE algorithm converges consistently.*

Proof: For a uniform distribution, $\delta > \frac{\epsilon}{2\sqrt{\gamma_*}}$. So, given $\epsilon > 0$, a positive δ is obtained by this inequality. Hence, convergence is guaranteed by Theorem 1. In the Bernoulli case, convergence is assured by Theorem 1 since the bound is visited at each step. □

4 Simulations

Consider an AR(3) model

$$y_n = a_1 y_{n-1} + a_2 y_{n-2} + a_3 y_{n-3} + v_n \tag{17}$$

where $a_1 = 2, a_2 = -1.48, a_3 = 0.34$ and v_n is a binary Bernoulli sequence, $B(-1,1)$, with

$$v_n = \{1, \text{ with probability } 0.7; \quad -1, \text{ with probability } 0.3\}. \tag{18}$$

The error bound ($\gamma_* = 1$) and the mean ($= 0.2$) of v_n are assumed *unknown*. Figure 1 shows the simulation result of OBE-ABE ($\gamma_0 = 1.5$, $\epsilon = 0$, and $N = 20$) and SM-SA

in which $\gamma_0 = 1.5$. Neither estimator is affected by the non-zero mean of v_n and both consistently converge to the true parameter ($a_1 = 2$). This is guaranteed by Lemma 2 and Theorem 1. Further, with the help of ABE, the rate of convergence of the OBE-ABE algorithm is faster than that of SM-SA (hence any OBE algorithm) as seen in the smaller ellipsoid volumes (Fig. 2). This also follows from Lemma 1 and Theorem 1.

OBE-ABE is also more robust to a noisy environment than conventional OBE algorithms. Consider Fig. 3 in which the input $\{v_n\}$ is a realization of an i.i.d. sequence with equally distributed discrete values {-1, -0.5, 0, 0.5, 1}, and the noise $\{w_n\}$ (independent of $\{v_n\}$) is i.i.d. uniform on [-0.05, 0.05] (S/N=42.5 dB). As shown in Fig. 5, the ellipsoid volumes of SM-SA ($\gamma_n = 1$) quickly become negative because of the added noise w_n (disintegration of the optimization process), while the volumes of OBE-ABE ($\gamma_0 = 10, \epsilon = 0.5$, and $N = 100$) quickly converge from above to zero. Figure 4 shows the first parameter estimate from both algorithms. While the estimate of SM-SA diverges, the estimate of OBE-ABE converges to the theoretical values ($[\hat{a}_1, \hat{a}_2, \hat{a}_3] = [1.9930, -1.4676, 0.3332]$) found by solving Wiener equation.

5 Conclusion

Conventional OBE algorithms exhibit many practical advantages including computational efficiency and excellent tracking ability, but these algorithms provide inconsistent estimates in the presence of improper bounds. Under-bounding can be catastrophic as the bounding ellipsoid may eventually exclude the true parameters. Using conservatively "loose" bounds prevents the exclusion of the true parameters from the feasible set, but prevents sufficient convergence and computational efficiency in many cases. Therefore, the ability to properly estimate bounds is of critical practical significance. The OBE-ABE method requires only a lower bound on the "tail probability" of the model error sequence. OBE-ABE provides superior performance relative to conventional OBE algorithms with conservatively estimated bounds in terms of convergence, speed of convergence, computational efficiency, and robustness to measurement noise. A proof of conditions for consistent convergence of OBE-ABE has been given.

References

[1] Dasgupta, S., and Y. F. Huang, "Asymptotically convergent modified recursive least squares with data dependent updating and forgetting factor for systems with bounded noise," *IEEE Trans. Info. Theory,* vol. 33, pp. 383–392, 1987.

[2] Deller Jr., J. R., M. Nayeri and S. F. Odeh, "Least-square identification with error bounds for real-time signal processing and control," *Proc. IEEE,* vol. 81, pp. 815–849, 1993.

[3] Deller, Jr., J. R., M. Nayeri, and M. S. Liu, "Unifying the landmark developments in optimal bounding ellipsoid identification," *Int. J. Adaptive Control and Signal Processing*, vol. 8, pp. 43–60, 1994.

[4] Deller, J.R., and T.C. Luk, "Linear prediction analysis of speech based on set-membership theory," *Computer Speech and Language*, vol. 3, pp. 301-327, 1989.

[5] Lin, T.M., M. Nayeri, and J.R. Deller, "A consistently convergent OBE algorithm with automatic estimation of error bounds," *Int. J. Adaptive Control and Signal Processing* (in review).

[6] Liu, M. S., *Development and Analysis of an Optimal Bounding Ellipsoid Algorithm Using Stochastic Approximation* (Ph.D. dissertation), Michigan State Univ., 1993.

[7] Milanese, M., H. Piet-Lahanier, J. Norton, and E. Walter (editors), *Bounding Approaches to System Identification*, London: Plenum, 1996.

[8] Nayeri, M., M.S. Liu, and J.R. Deller, Jr., "Do interpretable optimal bounding ellipsoid algorithms converge? Part I - The long-awaited set-convergence proof," *Proc. 10th IFAC Symp. on System Identification (SYSID '94)*, July, vol. 3, pp. 389-394, Copenhagen, 1994.

[9] Rao, A. K., Y. F. Huang, "Tracking characteristics of an OBE parameter estimation algorithm," *IEEE Trans. Signal Processing*, vol. 41, pp. 1140–1148, 1993.

[10] Söderström, T., and P. Stoica, *System Identification*, Englewood Cliffs, New Jersey: Prentice-Hall, 1989.

[11] Veres, S.M., and J.P. Norton, "Structure selection for bounded-parameter models: Consistency conditions and selection criterion," *IEEE Trans. Automatic Control*, vol. 36, pp. 474-481, 1991.

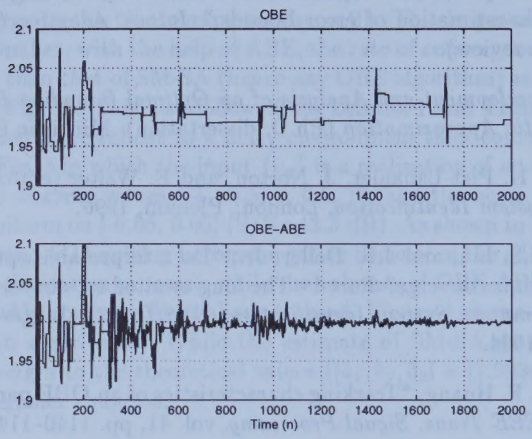


Figure 1: Estimates of a_1 by OBE and OBE-ABE with $v_n \sim B(-1,1)$ non-zero mean.

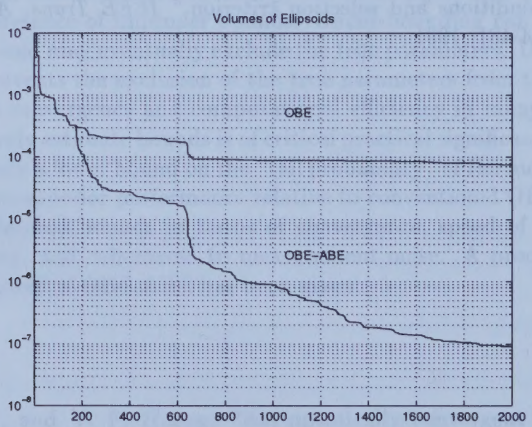


Figure 2: Volumes of the ellipsoids obtained by OBE and OBE-ABE with $v_n \sim B(-1,1)$ non-zero mean.

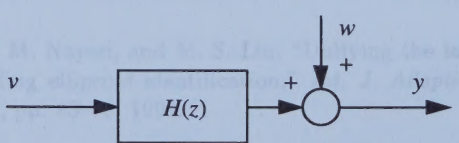


Figure 3: Block diagram of a noisy environment.

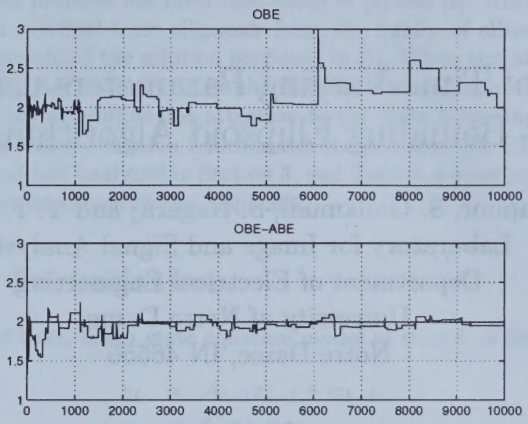


Figure 4: Estimates of a_1 by OBE and OBE-ABE applied to the system of Fig. 3 (S/N=42.5 dB) with $\gamma_n = 1$ for OBE, and $\gamma_0 = 10$ for OBE-ABE.

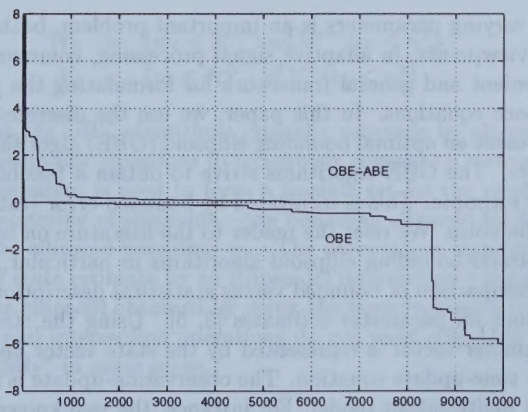


Figure 5: Ellipsoid volumes as OBE and OBE-ABE are applied to the system of Fig. 3 (S/N=42.5 dB) with $\gamma_n = 1$ for OBE, and $\gamma_0 = 10$ for OBE-ABE.

Tracking of Time-Varying Parameters using Optimal Bounding Ellipsoid Algorithms

S. Kapoor, S. Gollamudi, S. Nagaraj and Y. F. Huang
Laboratory for Image and Signal Analysis
Department of Electrical Engineering
University of Notre Dame
Notre Dame, IN 46556

Abstract

This paper describes an optimal bounding ellipsoid (OBE) estimation algorithm for tracking time-varying parameters with incrementally bounded time variations. A linear state-space model is used, with the time-varying parameters represented by the state vector. The OBE algorithm exhibits a selective update property for the time- and observation-update equations, and necessary and sufficient conditions for state tracking are derived. The interpretability of the optimization criterion is also investigated along with simulation results.

1 Introduction

Tracking of time varying parameters is an important problem, both from theoretical as well as practical viewpoints, in adaptive signal processing, communication and control systems. A convenient and general framework for formulating the problem is provided by linear state-space equations. In this paper, we use the discrete-time state equation framework and present an optimal bounding ellipsoid (OBE) algorithm for tracking time-varying parameters. The OBE algorithms strive to obtain a feasible set, rather than a unique parameter estimate. This is facilitated by bounded-error (rather than statistical) assumptions on the noise. We refer the reader to the literature on set-membership identification, and optimal bounding ellipsoid algorithms in particular, for the motivation, justification and comparison of bounded versus statistical description of the noise model and its ramifications on parameter estimates [3, 5]. Using the state-space model, the time-varying parameter vector is represented by the state vector and its time-evolution is modeled by the time-update equation. The observation-update is same as the conventional multi-input multi-output model. For instance, the well known first order Markov process model [7] used commonly for modeling time-varying parameter systems, is a special case of the state-space model.

The general problem of state estimation has been examined by numerous researchers for over three decades. The original state bounding algorithms are attributable to [13, 14]

and [1]. Recently, this problem has been considered in [4] and [9]. Also, [10] has derived minimal-volume and minimal-trace ellipsoids from the family of ellipsoids proposed in [14], and has also generalized the solution proposed in [5]. When statistical noise models are used, the Kalman filter is known to be the linear minimum variance state estimator. In this paper, however, we confine our attention to the state bounding approach. This paper is organized as follows: Section 2 formulates the problem; An OBE algorithm and its features are derived and analyzed in Section 3, and Section 4 presents some simulation results. All proofs are contained in the Appendix.

2 Problem Formulation

Consider the discrete-time, linear state equation model at time k of the form,

$$\begin{aligned} x_k &= A_{k-1}x_{k-1} + w_{k-1} \\ y_k &= C_k x_k + v_k \end{aligned} \tag{1}$$

where
x_k: $N \times 1$ state vector representing the time-varying parameter vector to be estimated
y_k: $L \times 1$ observation vector
A_k and C_k: Known matrices of dimensions $N \times N$ and $L \times N$ respectively
w_k: $N \times 1$ noise vector in time-update equation
v_k: $L \times 1$ noise vector in observation-update equation
All quantities are assumed real for convenience, and the extension to complex variables is straightforward. It is assumed here that w_k and v_k belong to ellipsoidal sets for all k according to,

$$\begin{aligned} W_k &= \{w_k \in \mathcal{R}^N : w_k^T W^{-1} w_k \leq \gamma_w^2\} \\ V_k &= \{v_k \in \mathcal{R}^N : v_k^T V^{-1} v_k \leq \gamma_v^2\} \end{aligned} \tag{2}$$

Parameter estimation via OBE algorithms typically proceeds by alternatively using the time- and observation-update equations to recursively compute a state estimate. At time k, the time-update equation is used to form a feasible set for the predicted state. This is done by a vector summing of ellipsoids – the bounding ellipsoid for the state estimate at time $k-1$ and the ellipsoid bounding w_k. The observation equation is then used to update the predicted state estimate by an ellipsoidal intersection – the ellipsoid from the previous step and the ellipsoid obtained by using the boundedness of v_k. In general, the ellipsoidal summing and intersection operations do not yield ellipsoids and thus have to be overbounded "tightly" in some sense.

3 An OBE Algorithm and Tracking Characteristics

Let

$$E(\hat{x}_k, P_k, \sigma_k^2) = \{x \in \mathcal{R}^N : [x - \hat{x}_k]^T P_k^{-1} [x - \hat{x}_k] \leq \sigma_k^2\} \tag{3}$$

denote the bounding ellipsoid at time k where $P_k > 0$, $\hat{x} \in \mathcal{R}^N$ and $\sigma_k^2 > 0$. Let the initial ellipsoid $E(\hat{x}_0, P_0, \sigma_0^2)$ be known. The time-update at time k is carried out by first linearly transforming the ellipsoid at time $k-1$, $E(\hat{x}_{k-1}, P_{k-1}, \sigma_{k-1}^2)$ to the ellipsoid $E(A_{k-1}\hat{x}_{k-1}, A_{k-1}P_{k-1}A_{k-1}^T, \sigma_{k-1}^2)$. This is followed by a vector sum of the resulting ellipsoid and W_{k-1} to yield an overbounding ellipsoid $E(\hat{x}_{k/k-1}, P_{k/k-1}, \sigma_{k/k-1}^2)$. It is notable that similar to the *selective observation-update*, the ellipsoidal summing may not be required at each k, leading to *selective time-update*. The necessary and sufficient conditions for selective time-update are presented in Section 3.2.

3.1 Observation Update

Let $\hat{x}_{k/k-1} = A_{k-1}\hat{x}_{k-1}$, $P_{k/k-1} = A_{k-1}P_{k-1}A_{k-1}^T$ and $\sigma_{k/k-1}^2 = \sigma_{k-1}^2$. The observation update at time k is carried out by seeking the intersection of $E(\hat{x}_{k/k-1}, P_{k/k-1}, \sigma_{k/k-1}^2)$ and S_k defined by

$$S_k = \{x \in \mathcal{R}^N : (y_k - C_k x)^T V^{-1} (y_k - C_k x) \leq \gamma^2\}$$

where $\gamma^2 > \gamma_v^2$. The motivation behind selecting γ^2 in this manner is essentially to improve the tracking performance of the algorithm (see Theorem 3, Section 3.2). Decomposing $V^{-1} = V_e V_e^T$, the above can be rewritten as

$$S_k = \{x \in \mathcal{R}^N : (y_k' - C_k' x)^T (y_k' - C_k' x) \leq \gamma^2\}$$

where $y_k' = V_e^T y_k$ and $C_k' = V_e^T C_k$. This is now a convenient form for deriving a recursive observation update algorithm. Let $\| \cdot \|$ denote the l_2 norm of a vector. Then, an ellipsoid that contains $E(\hat{x}_{k/k-1}, P_{k/k-1}, \sigma_{k/k-1}^2) \cap S_k$ is given by

$$\begin{aligned} E(\hat{x}_k, P_k, \sigma_k^2) \;=\; & \{x \in \mathcal{R}^N : (1-\lambda_k)[x - \hat{x}_{k/k-1}]^T P_{k/k-1}^{-1} [x - \hat{x}_{k/k-1}] \\ & + \lambda_k \| (y_k' - C_k' x) \|^2 \leq (1-\lambda_k)\sigma_{k/k-1}^2 + \lambda_k \gamma^2\} \end{aligned} \tag{4}$$

where λ_k is a real number in $[0, 1]$. We now show that there exists a symmetric positive definite matrix P_k and a positive σ_k^2 such that

$$E(\hat{x}_k, P_k, \sigma_k^2) = \{x \in \mathcal{R}^N : [x - \hat{x}_k]^T P_k^{-1} [x - \hat{x}_k] \leq \sigma_k^2\} \tag{5}$$

is a well defined ellipsoid [9].

Theorem 1. Consider the inequalities (4) and (5) above. Denoting $\delta_k \triangleq y_k' - C_k' \hat{x}_{k/k-1}$, $G_k \triangleq C_k' P_{k/k-1} C_k'^T$ and $Q_k \triangleq (1-\lambda_k) I_L + \lambda_k G_k$, we have the following recursive update equations,

$$P_k = \frac{1}{1-\lambda_k} [P_{k/k-1} - \lambda_k P_{k/k-1} C_k'^T Q_k^{-1} C_k' P_{k/k-1}], \tag{6}$$

$$\hat{x}_k = \hat{x}_{k/k-1} + \lambda_k P_k C_k'^T \delta_k \qquad \text{and} \tag{7}$$

$$\sigma_k^2 = (1-\lambda_k)\sigma_{k/k-1}^2 + \lambda_k \gamma^2 - (1-\lambda_k)\lambda_k \delta_k^T Q_k^{-1} \delta_k \qquad \blacksquare$$

The proof follows straightforwardly from (4) and (5) and is given in the Appendix. The update parameter λ_k is optimized at each step by minimizing a tight upper bound on σ_k^2 which can be considered to be a bound on the estimation error, or as a scaling factor of the matrix P_k at time k [2]. An upper bound on σ_k^2 denoted by $\sigma_k'^2$, is given by

$$\sigma_k'^2 = (1-\lambda_k)\sigma_{k/k-1}^2 + \lambda_k\gamma^2 - (1-\lambda_k)\lambda_k \frac{\delta_k^T\delta_k}{(1-\lambda_k)+\lambda_k g_k} \tag{8}$$

where g_k is the 2-norm of G_k. Denoting the optimal λ_k by λ_k^o (which lies in $[0,\alpha]$ for some $\alpha < 1$) and defining the quantity $\beta_k \triangleq (\gamma^2 - \sigma_{k/k-1}^2)/\delta_k^T\delta_k$, we have,

Theorem 2. Minimization of $\sigma_k'^2$ with respect to λ_k, leads to the following update condition:
(1) if $\beta_k \geq 1$, then $\lambda_k^o = 0$
(2) else $\lambda_k^o = \min(\alpha, \nu_k)$ where $\nu_k = \begin{cases} \alpha & \text{if } \delta_k^T\delta_k = 0 \\ \frac{(1-\beta_k)}{2} & \text{if } g_k = 1 \\ \frac{1}{1-g_k}[1 - \sqrt{\frac{g_k}{1+\beta_k[g_k-1]}}] & \text{if } 1+\beta_k[g_k-1] > 0 \\ \alpha & \text{if } 1+\beta_k[g_k-1] \leq 0 \end{cases}$ ∎

The above result follows straightforwardly from the derivation given in [2]. We see that the discerning update criterion involves the computation of only the prediction error, δ_k.

3.2 Time Update

Theorem 2 results in a selective observation-update which can be exploited for significant computational savings [2, 3, 5, 6]. The following results demonstrate the conditions under which the OBE algorithm successfully tracks time variations in the state vector. The true state $x_k \in E(\hat{x}_k, P_k, \sigma_k^2)$ if $x_k \in E(\hat{x}_{k/k-1}, P_{k/k-1}, \sigma_{k/k-1}^2)$. Thus, a sufficient condition for tracking the true state is

$$(\hat{x}_{k/k-1} - x_k)^T P_{k/k-1}^{-1}(\hat{x}_{k/k-1} - x_k) \leq \sigma_{k/k-1}^2$$

which can be rewritten as,

$$(\hat{x}_{k-1} - A_{k-1}^{-1}x_k)^T P_{k-1}^{-1}(\hat{x}_{k-1} - A_{k-1}^{-1}x_k) \leq \sigma_{k-1}^2$$

Further, the true state can belong to a region outside the bounding ellipsoid without loss of tracking. This region is described by the following:
Theorem 3. The true state $x_k \in E(\hat{x}_k, P_k, \sigma_k^2)$ if and only if,

$$(\hat{x}_{k-1} - A_{k-1}^{-1}x_k)^T P_{k-1}^{-1}(\hat{x}_{k-1} - A_{k-1}^{-1}x_k) \leq \sigma_{k-1}^2 + \frac{\lambda_k}{1-\lambda_k}(\gamma^2 - \| v_{k-1} \|^2)$$ ∎

Thus, increasing σ_{k-1}^2, the update parameter λ_k, or γ^2 can enhance the tracking capability. More specifically, the following sufficient condition emerges straightforwardly from above.

Theorem 4. If $x_{k-1} \in E(\hat{x}_{k-1}, P_{k-1}, \sigma_{k-1}^2)$ and $\lambda_k \neq 0$, then $x_k \in E(\hat{x}_k, P_k, \sigma_k^2)$ if,

$$\| w_{k-1} \| \leq \frac{1}{\lambda_{max}(A_{k-1}^{-1})\sqrt{\lambda_{min}(P_{k-1}^{-1})}} \left[\sqrt{\sigma_{k-1}^2 + \frac{\lambda_k}{1-\lambda_k}(\gamma^2 - \gamma_v'^2)\frac{\lambda_{min}(P_{k-1}^{-1})}{\lambda_{max}(P_{k-1}^{-1})}} - \sqrt{\sigma_{k-1}^2} \right] \quad (9)$$

where $\lambda_{max}(A_{k-1}^{-1})$ is the maximum singular value of A_{k-1}^{-1}, $\lambda_{min}(P_{k-1}^{-1})$ and $\lambda_{max}(P_{k-1}^{-1})$ are, respectively, the minimum and maximum singular values of P_{k-1}^{-1} and $\gamma_v'^2$ is the actual bound on the observation noise. In order to evaluate (9), the left hand side may be replaced by γ_w. If Theorem 4 is satisfied at time k, there is no need for an ellipsoid summing operation resulting in selective time-update. Since Theorem 4 is merely a sufficient condition for successful tracking, its use will result in a conservative tracking strategy. Nevertheless, it yields further insight into the tracking behavior of the OBE algorithm.

The above analysis sheds light on the behavior of the OBE algorithm when the parameter vector is time-varying. It also explains the basis for various schemes proposed in the literature for tracking. For instance, Schweppe [14] proposed a scheme to add a positive definite matrix to P_k for all k. This has the effect of increasing the eigenvalues of P_k and thus increasing the right hand side of (9), making the sufficient condition easier to satisfy. Rao and Huang [12] carried out a detailed analysis of the parameter tracking problem for the single-input single-output case and proposed an increase in σ_k^2 to prevent the ellipsoid from shrinking to zero. Also, simply using a larger than necessary noise bound is seen to be beneficial. However, this benign effect has to be traded off with a possible enhancement in estimation error.

3.3 Interpretability of σ_k^2 minimization

We now investigate the interpretability of σ_k^2 minimization, *i.e.*, its relation to minimization of interpretable measures like determinant and trace of the positive definite matrix $\sigma_k^2 P_k$ which are directly related to the size of the bounding ellipsoid. These measures are proportional to the volume and sum of semi-axes of the bounding ellipsoid, respectively. The issue of relating σ_k^2 minimization to determinant or trace minimization has also been addressed in [3]. It however, deals primarily with the relation between the update criterion resulting from the different minimization procedures. Define

$$B_k \triangleq \sigma_k^2 P_k, \quad (10)$$

and denote the determinant and trace of B_k by [5]

$$\begin{aligned} \mu_k^{vol} &\triangleq det(B_k) \\ \mu_k^{tr} &\triangleq tr(B_k) \end{aligned} \quad (11)$$

Also consider the following determinant and trace measures [5, 3]

$$\begin{aligned} \nu_k^{vol} &\triangleq \mu_k^{vol}/\mu_{k/k-1}^{vol} = det(B_{k/k-1}^{-1} B_k) \\ \nu_k^{tr} &\triangleq \mu_k^{tr}/\mu_{k/k-1}^{tr} = tr(B_k)/tr(B_{k/k-1}) \end{aligned} \quad (12)$$

Theorem 5. By minimizing σ_k^2 for computing the optimal observation update factor λ_k, the following relations hold for the ellipsoidal determinant and trace measures defined in (11) and (12):

1. $\mu_k^{vol} \leq c_v(\sigma_k^2)^N$ for some constant c_v
2. $\mu_k^{tr} \leq c_t\sigma_k^2$ for some constant c_t.
   Also, for $L = 1$,
3. $\nu_k^{vol} \leq [\sigma_k^2/((1-\lambda_k)\sigma_{k/k-1}^2)]^N$
4. $\nu_k^{tr} \leq \sigma_k^2/((1-\lambda_k)\sigma_{k/k-1}^2)$ ∎

The upper bounds in items (3) and (4) become exact when $\lambda_k = 0$ (see the Appendix). Since we know that $\lambda_k \to 0$ with D-H/OBE, the upper bounds in (3) and (4) are asymptotically exact. Theorem 5 reveals that minimizing $\sigma_k^2\ \forall\ k$ results in minimizing upper bounds on interpretable ellipsoidal volume and trace measures. These results show that although using σ_k^2 minimization, the determinant and trace (11) are not monotonically decreasing (as is the case for the more common determinant and trace minimization OBE algorithms [5, 3, 10]), μ_k^{vol} and μ_k^{tr} are upper bounded by monotonically decreasing upper bounds at each step.

4 Simulations

Consider a first order Markov process model [7] with $N = 3$ and $L = 2$; $A_k = aI_N$ for all k and $a = 0.98$; each component of w_k and v_k is taken to be uniformly distributed in the intervals $[-0.02, 0.02]$ and $[-0.05, 0.05]$ respectively; C_k is drawn from a N dimensional Gaussian random process. Figure 1(a) shows a plot of μ_k^{vol}, μ_k^{tr} and σ_k^2 versus time for a typical simulation run. The bump in all the indicators near time $k = 80$ is because of increasing σ_k^2 by 0.25 to prevent it from becoming negative due to drifting of the parameter being tracked. The mean square error of the parameter estimate for the first 50 time instants is averaged over 100 independent trials and is plotted in Figure 1(b).

5 Conclusions

This paper has considered the problem of tracking time-varying parameters in a state-space framework using an OBE algorithm. OBE algorithms are well suited for time-varying parameter estimation due to their inherent structure and mathematical tractability. An OBE algorithm and its tracking features have been presented along with an investigation on the interpretability of σ_k^2 minimization criterion.

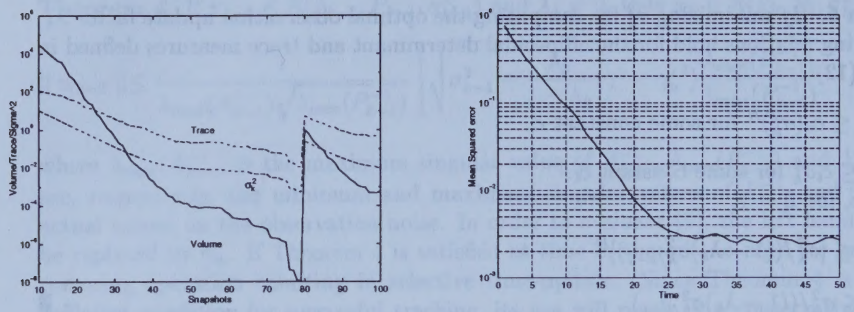


Figure 1: (a) Plotting μ_k^{vol},μ_k^{tr} and σ_k^2 versus time (b) MSE

Appendix

Proof of Theorem 1: To obtain the update equation for P_k, we equate he quadratic terms of (4) and (5) to yield,

$$P_k^{-1} = (1-\lambda_k)P_{k/k-1}^{-1} + \lambda_k C_k'^T C_k'$$

Using the matrix inversion lemma, we have

$$P_k = \frac{1}{1-\lambda_k}[P_{k/k-1} - P_{k/k-1}C_k'^T[(1-\lambda_k)I_L + \lambda_k C_k' P_{k/k-1} C_k'^T]^{-1}\lambda_k C_k' P_{k/k-1}] \quad (13)$$

Defining Q_k as in (8), (13) reduces to (6). Similarly, equating the linear terms of (4) and (5) yields,

$$P_k^{-1}\hat{x}_k = (1-\lambda_k)P_{k/k-1}^{-1}\hat{x}_{k/k-1} + C_k'^T\lambda_k y_k'$$

Using (13), this can be simplified to,

$$\hat{x}_k = \hat{x}_{k/k-1} + P_{k/k-1}C_k'^T Q_k^{-1}\lambda\delta_k$$

Now substituting Q_k and simplifying, (7) is obtained. Finally, equating the constant terms gives,

$$\sigma_k^2 - \hat{x}_k^T P_k^{-1}\hat{x}_k = (1-\lambda_k)\sigma_{k/k-1}^2 + \lambda_k\gamma^2 - (1-\lambda_k)\hat{x}_{k/k-1}^T P_{k/k-1}^{-1}\hat{x}_{k/k-1} - \lambda_k y_k'^T y_k' \quad (14)$$

Using the definitions above, collecting terms, and after some straightforward algebra, (14) above reduces to the desired expression. ∎

Sketch of Proof of Theorem 3: The necessary and sufficient condition for $x_k \in E(\hat{x}_k, P_k, \sigma_k^2)$ is

$$(x_k - \hat{x}_k)^T P_k^{-1}(x_k - \hat{x}_k) \leq \sigma_k^2 \quad (15)$$

Using (1) and (13), along with extended manipulations, we obtain

$$(x_k - \hat{x}_k)^T P_k^{-1}(x_k - \hat{x}_k) - \sigma_k^2 = (1-\lambda_k)[z_k^T P_{k/k-1}^{-1} z_k - \sigma_k^2] + \lambda_k[|| v_k ||^2 - \gamma^2]$$

where $z_k = A_{k-1}\hat{x}_{k-1} - x_k$ for convenience. Using (15), and some rearrangement, the desired expression is obtained. ∎

Sketch of Proof of Theorem 4: Using (1) and Theorem 3, we have

$$\begin{aligned}\tilde{x}_{k-1}^T P_{k-1}^{-1}\tilde{x}_{k-1} - 2\tilde{x}_{k-1}^T P_{k-1}^{-1}A_{k-1}^{-1}w_{k-1} \;+\; & (A_{k-1}^{-1}w_{k-1})^T P_{k-1}^{-1}A_{k-1}^{-1}w_{k-1} \\ \leq\; & \sigma_{k-1}^2 + \frac{\lambda_k}{1-\lambda_k}(\gamma^2 - \| v_{k-1} \|^2)\end{aligned}$$

where $\tilde{x}_k = (\hat{x}_k - x_k)$ for convenience. Assuming that $x_{k-1} \in E(\hat{x}_{k-1}, P_{k-1}, \sigma_{k-1}^2)$, a sufficient condition for the above is

$$\begin{aligned}2 \| \tilde{x}_{k-1} \| \lambda_{max}(P_{k-1}^{-1})\lambda_{max}(A_{k-1}^{-1}) \| w_{k-1} \| \;+\; & \| w_{k-1} \|^2 \lambda_{max}(P_{k-1}^{-1})\lambda_{max}^2(A_{k-1}^{-1}) \\ \leq\; & (\gamma^2 - \| v_{k-1} \|^2)\end{aligned}$$

Again, since $x_{k-1} \in E(\hat{x}_{k-1}, P_{k-1}, \sigma_{k-1}^2)$, $\| \tilde{x}_{k-1} \|^2 \leq \sigma_{k-1}^2/\lambda_{min}(P_{k-1}^{-1})$. Substituting and solving the quadratic for $\| w_{k-1} \|$ yields the desired result. ∎

Proof of Theorem 5:

1. It can be shown that under a persistence of excitation assumption of the input data, i.e., if there exist N, α_1, α_2 such that

$$0 < \alpha_1 I \leq \sum_{i=k}^{k+N} x_i x_i^T \leq \alpha_2 I \quad \forall k,$$

then the matrix P_k is upper bounded [2] as $P_k \leq \eta I_N$, for some constant η. For instance, $\eta = \sup_k\{\lambda_{max}(P_k)\}$. Thus,

$$det(B_k) = (\sigma_k^2)^N det(P_k) \leq c_v(\sigma_k^2)^N$$

with $c_v = \eta^N$. ∎

2. Similar to item 1 above,

$$tr(B_k) = \sigma_k^2 tr(P_k) \leq c_t\sigma_k^2$$

with $c_t = N\eta$. ∎

3. Using (10) and (13) it follows that,

$$\frac{B_k}{\sigma_k^2} = \frac{1}{1-\lambda_k}\left[\frac{B_{k/k-1}}{\sigma_{k/k-1}^2} - \frac{\lambda_k B_{k/k-1}C_k'^T C_k' B_{k/k-1}}{(\sigma_{k/k-1}^2)^2(1-\lambda_k+\lambda_k G_k)}\right]$$

Using (12),

$$\nu_k^{vol} = det\left(\frac{(\sigma_k^2/\sigma_{k/k-1}^2)}{1-\lambda_k} I_N - \frac{\lambda_k\sigma_k^2 C_k'^T C_k' B_{k/k-1}}{(1-\lambda_k)(\sigma_{k/k-1}^2)^2(1-\lambda_k+\lambda_k G_k)}\right)$$

Using the identity $det(aI + bc^T) = a^{N-1}(a + c^T b)$ where a is a real constant and $b, c \in \mathcal{R}^{N\times 1}$, we have,

$$\nu_k^{vol} = \left[\frac{(\sigma_k^2/\sigma_{k/k-1}^2)}{1-\lambda_k}\right]^{N-1}\left[\frac{(\sigma_k^2/\sigma_{k/k-1}^2)}{1-\lambda_k} - \frac{\lambda_k\sigma_k^2 C_k' B_{k/k-1}C_k'^T}{(1-\lambda_k)(\sigma_{k/k-1}^2)^2(1-\lambda_k+\lambda_k G_k)}\right]$$

which reduces to

$$\nu_k^{vol} = \left[\frac{(\sigma_k^2/\sigma_{k/k-1}^2)}{1-\lambda_k}\right]^N \left[1 - \frac{\lambda_k G_k}{1-\lambda_k+\lambda_k G_k}\right]$$

Since P_k is symmetric positive definite, it follows that,

$$\nu_k^{vol} \leq \left[\frac{\sigma_k^2}{(1-\lambda_k)\sigma_{k/k-1}^2}\right]^N \tag{16}$$

Error in the upper bound is given by,

$$\Delta = \left[\frac{\sigma_k^2}{\sigma_{k/k-1}^2}\right]^N \left[\frac{1}{(1-\lambda_k)^N} - \frac{1-\lambda_k}{(1-\lambda_k)^N(1-\lambda_k+\lambda_k G_k)}\right]$$

As per the convergence of the OBE algorithm, as $k \to \infty$, $\lambda_k \to 0$, thus making the bound asymptotically exact. ∎

4. Using (12), we have

$$tr(B_k) = \frac{(\sigma_k^2/\sigma_{k/k-1}^2)}{1-\lambda_k} tr(B_{k/k-1}) - tr\left(\frac{\lambda_k \sigma_k^2 B_{k/k-1} C_k'^T C_k' B_{k/k-1}}{(1-\lambda_k)(\sigma_{k/k-1}^2)^2(1-\lambda_k+\lambda_k G_k)}\right)$$

Using the identity $tr(xy^T) = (x^T y)$,

$$\nu_k^{tr} = \frac{(\sigma_k^2/\sigma_{k/k-1}^2)}{1-\lambda_k} - \frac{1}{tr(B_{k/k-1})} \frac{\lambda_k \sigma_k^2 C_k' B_{k/k-1}^2 C_k'^T}{(1-\lambda_k)(\sigma_{k/k-1}^2)^2(1-\lambda_k+\lambda_k G_k)}$$

Rewriting,

$$\nu_k^{tr} = \frac{(\sigma_k^2/\sigma_{k/k-1}^2)}{1-\lambda_k}\left[1 - \frac{\lambda_k H_k}{(1-\lambda_k)(1-\lambda_k+\lambda_k G_k)T_{k-1}}\right]$$

where $H_k \triangleq C_k' P_{k-1}^2 C_k'^T$ and $T_{k-1} \triangleq tr(P_{k-1})$. Since P_k is symmetric positive definite, we obtain,

$$\nu_k^{tr} \leq \frac{(\sigma_k^2/\sigma_{k/k-1}^2)}{1-\lambda_k}$$

Similar to (16), the asymptotic exactness of the upper bound on ν_k^{tr} is obtained. ∎

References

[1] D. P. Bertsekas and I. B. Rhodes, "Recursive state estimation for a set-membership description of uncertainty," *IEEE Trans. on Automatic Control*, Vol. 16, pp. 117-128, April 1971.

[2] S. Dasgupta and Y. F. Huang, "Asymptotically Convergent Modified Recursive Least Squares with Data Dependent Updating and forgetting factor for Systems with Bounded Noise," *IEEE Transactions on Information Theory*, Vol. 33, No. 3, pp. 383-392, May 1987.

[3] J. R. Deller, M. Nayeri and S. F. Odeh, "Least square identification with error bounds for real-time signal processing and control," *Proc. IEEE*, Vol. 81, No. 6, pp. 813-849, June 1993.

[4] C. Durieu, B. T. Polyak and E. Walter, "Ellipsoidal State Outer-Bounding for MIMO Systems via Analytical Techniques," *Proc., Symp. Modeling, Analysis and Simulation, CESA'96 IMACS Multiconference*, pp. 843-848, July, 1996.

[5] E. Fogel and Y. F. Huang, "On the value of information in system identification - bounded noise case," *Automatica*, Vol. 18, No. 2, pp. 229-238, March 1982.

[6] S. Gollamudi and Y.F. Huang, "Updater-Shared Adaptive Parallel Equalization (U-SHAPE) using set-membership identification," *Proc. IEEE Intl. Conf. Circuits and Systems*, Atlanta, GA, USA, May 13-15, 1996.

[7] S. Haykin, *Adaptive Filter Theory*, Prentice Hall, 1996.

[8] Y.F. Huang and J. R. Deller, "On the tracking capabilities of optimal bounding ellipsoid algorithms," *Proc. 30th Ann. Allerton Conf. on Comm., Control and Comp.*, pp. 50-59, October 1992.

[9] S. Gollamudi, S. Nagaraj, S. Kapoor and Y.F. Huang, "Set Membership State Estimation with Optimal Bounding Ellipsoids," *Proc. Intl. Symp. on Information Theory and its App.*, pp. 262-265, Victoria, B.C., Canada, September 17-20, 1996.

[10] D. G. Maksarov and J. P. Norton, "State Bounding with Ellipsoidal Set Description of the Uncertainty," *Int. Journal of Control*, September 1996.

[11] J. P. Norton and S. H. Mo, "Parameter bounding for time-varying systems," *Mathematics and Computers in Simulation*, Vol. 32, pp. 527-534, 1990.

[12] A. K. Rao and Y. F. Huang, "Tracking Characteristics of an OBE Parameter Estimation Algorithm," *IEEE Transactions on Signal Processing*, Vol. 41, No. 3, pp. 1140-1148, March 1993.

[13] F. M. Schlaepfer and F. C. Schweppe, "Continuous time state estimation under disturbances bounded by convex sets," *IEEE Trans. on Automatic Control*, Vol. 17, pp. 197-205, 1972.

[14] F. C. Schweppe, "Recursive state estimation: Unknown but bounded errors and system inputs," *IEEE Trans. on Automatic Control*, Vol. 13, pp. 22-28, 1968.

A Simple Way to Adjust the LMS Algorithm Step Size

Yuadkoun Dhiantravan and Roland Priemer
Department of Electrical Engineering and Computer Science
University of Illinois at Chicago
Chicago, Illinois

Abstract : This paper proposes a simple method to speed up convergence of the LMS algorithm by automatically adjusting the step size. The proposed method retains the simplicity of the conventional LMS algorithm.

INTRODUCTION

The Least Mean Square (LMS) algorithm possesses many preferred properties compared to other adaptive algorithms. Some of these properties are simplicity, robustness, and stability. Still, it has some drawbacks such as slow convergence and selection of a suitable step size [1, 2]. The step size is the trade-off factor between the convergence rate and the steady state error.

Numerous research efforts have been directed to improve the LMS algorithm convergence rate. Some achieve a better convergence rate by using a momentum term that is added to the conventional LMS algorithm [3, 4]. Many improve convergence by adjusting the step size [5, 6, 7].

However, most algorithms need to find suitable parameters that are introduced by the new algorithms to achieve better performance. Some [7] show that by introducing new parameters, a faster convergence rate with lower steady state error than with the conventional LMS algorithm can be achieved. The method given here also adjusts the step size. This is based on trends of prediction errors, and it does not introduce new parameters to be adjusted.

DEFINITIONS AND BASIC EQUATIONS

Assume that the scalar signal $y(n)$ depends on the scalar data $u(n)$, and consider modeling the signal $y(n)$ with a linear combination of M data points, $u(n-i)$, $i = 0,\ldots,M-1$, by

$$y(n) = \sum_{i=0}^{M-1} a_i u(n-i) + w(n) \tag{1}$$

where the a_i are a set of coefficients and $w(n)$ is zero mean white Gaussian noise with variance σ_w^2. The LMS algorithm is concerned with estimating the set of M parameters h_i, $i = 0,\ldots,M-1$, that minimize the mean square error. The minimum mean square modeling error $\varepsilon(n)$ is given by

$$\varepsilon(n) = y(n) - \mathbf{U}^{\mathrm{T}}(n)\mathbf{H} \tag{2}$$

where T denotes the matrix transpose, $\mathbf{U}(n)$ is a collection of M data points given by $\mathbf{U}^{\mathrm{T}}(n) = \left[u(n), u(n-1),\ldots,u(n-(M-1))\right]$ and $\mathbf{H}$ is the collection of M parameters $\mathbf{H}^{\mathrm{T}} = \left[h_0, h_1,\ldots,h_{M-1}\right]$, which is called the true coefficient vector. As in [1], we assume that $\varepsilon(n)$ is a stationary white noise process with zero mean and variance σ_ε^2.

Let $\hat{\mathbf{H}}(n)$ denote an estimate of $\mathbf{H}$ determined with the data vector sequence $\mathbf{U}(0), \mathbf{U}(1),\ldots,\mathbf{U}(n)$. Then $e(n/n-1)$ as given by

$$e(n/n-1) = y(n) - \mathbf{U}^{\mathrm{T}}(n)\hat{\mathbf{H}}(n-1) \qquad ; n = 0,1,\ldots \tag{3}$$

is called the a-priori estimation error, and $e(n/n)$ as given by

$$e(n/n) = y(n) - \mathbf{U}^{\mathrm{T}}(n)\hat{\mathbf{H}}(n) \tag{4}$$

is called the a-posteriori estimation error.

The LMS algorithm uses the instantaneous estimate of the gradient of the mean square error (MSE) to update the estimated parameter vector, i.e.

$$\hat{\mathbf{H}}(n) = \hat{\mathbf{H}}(n-1) + \mu \mathbf{U}(n) e(n/n-1) \tag{5}$$

where $\frac{\mu}{2}$ is the step size. From eqs. (3), (4) and (5) we obtain

$$e(n/n) = e(n/n-1)\left[1 - \mu \mathbf{U}^{\mathrm{T}}(n)\mathbf{U}(n)\right] \tag{6}$$

The stability of the algorithm is fulfilled by satisfying

$$\left|1 - \mu \mathbf{U}^{\mathrm{T}}(n)\mathbf{U}(n)\right| < 1 \tag{7}$$

As the algorithm proceeds, the average estimation error is minimized in the mean square sense. Therefore the ensemble statistics of the absolute value of two consecutive errors are constrained to

$$\frac{\mathbf{E}\left[\left|e(n/n)\right|\right]}{\mathbf{E}\left[\left|e(n-1/n-1)\right|\right]} < 1 \tag{8}$$

From eqs. (6) and (8) we obtain

$$\frac{\mathbf{E}\left[\left|e(n/n)\right|\right]}{\mathbf{E}\left[\left|e(n-1/n-1)\right|\right]} = \frac{\mathbf{E}\left[\left|e(n/n-1)\left[1 - \mu \mathbf{U}^{\mathrm{T}}(n)\mathbf{U}(n)\right]\right|\right]}{\mathbf{E}\left[\left|e(n-1/n-2)\left[1 - \mu \mathbf{U}^{\mathrm{T}}(n-1)\mathbf{U}(n-1)\right]\right|\right]} < 1 \tag{9}$$

ADJUSTING THE STEP SIZE

We replace μ with $\mu(n)$, which represents the step size for iteration n, and rewrite eqs.(7) and (9) respectively as

$$\left|1 - \mu(n)\mathbf{U}^{\mathrm{T}}(n)\mathbf{U}(n)\right| = p_n \tag{10}$$

where $0 < p_n < 1$, and

$$\frac{\mathbf{E}\left[\left|e(n/n)\right|\right]}{\mathbf{E}\left[\left|e(n-1/n-1)\right|\right]} = \frac{\mathbf{E}\left[\left|e(n/n-1)\left[1-\mu(n)\mathbf{U}^{\mathrm{T}}(n)\mathbf{U}(n)\right]\right|\right]}{\mathbf{E}\left[\left|e(n-1/n-2)\left[1-\mu(n-1)\mathbf{U}^{\mathrm{T}}(n-1)\mathbf{U}(n-1)\right]\right|\right]} = q_n \quad (11)$$

where $0 < q_n < 1$. With eqs.(10) and (11) we have

$$\frac{\mathbf{E}\left[\left|e(n/n-1)\left[1-\mu(n)\mathbf{U}^{\mathrm{T}}(n)\mathbf{U}(n)\right]\right|\right]}{\mathbf{E}\left[\left|e(n-1/n-2)p_{n-1}\right|\right]} = q_n \quad (12)$$

After some manipulations eq.(12) and using instantaneous errors instead of the ensemble statistics, we write the equation to adjust the step size as

$$\mu(n) = \frac{1}{\mathbf{U}^{\mathrm{T}}(n)\mathbf{U}(n)}\left[1 - p_{n-1}q_n\frac{\left|e(n-1/n-2)\right|}{\left|e(n/n-1)\right|}\right] \quad (13)$$

Initially, the values of p and q are assigned to be between zero and one. As the algorithm proceeds, we let $p_n = p_{n-1}$ and examine the term in brackets of eq.(13). If it is negative, we decrease the value of p_{n-1} by a factor of d such as $p_{n-1} = dp_{n-1}$, where the decrease factor d satisfies $0 < d < 1$.

REVIEW OF TWO ALGORITHMS

In this section we review the algorithms by Harris [5] and Kwong and Johnston [7]. In [5], the algorithm uses a different step size for each weight. The weight is updated according to the equation

$$\hat{\mathbf{H}}_i(n) = \hat{\mathbf{H}}_i(n-1) + \mu_i(n)\mathbf{U}_i(n)e(n) \quad (14)$$

The parameter $\mu_i(n)$ is updated as follows:

1) $\mu_i(n) = \mu_i(n-1)/\alpha$ if there are m_0 consecutive sign changes in $\mathbf{U}_i(n)e(n)$

2) $\mu_i(n) = \alpha\,\mu_i(n-1)$ if there are m_1 consecutive identical signs in $\mathbf{U}_i(n)e(n)$

The $\mu_i(n)$ are allowed to vary between the values of μ_{max} and μ_{min} with the constraint of $1/\lambda > \mu_{max} > 0$ where λ is the maximum eigenvalue of the input correlation matrix. The minimum μ_{min} is chosen to provide the desired misadjustment after the algorithm has converged. The initial step size parameters are typically set equal to μ_{max}. Also, $\alpha > 1$, and usually it is assigned to $\alpha = 2$. For m_0 and m_1, [5] recommends $m_0 = m_1 = 2$ or 3 and $m_0 = m_1 - 1 = 2$ for a highly contaminated signal. Nevertheless, [7] compared its algorithm with [5], and used the values of m_0 and m_1 from $m_0 = m_1 = 3$ to $m_0 = m_1 - 1 = 2$ and 4. In [7] the variable step size is based on the mean square error at every iteration. The step size parameter is updated with

$$\mu'(n) = \alpha\mu(n-1) + \gamma e^2(n/n-1) \tag{15}$$

where $0 < \alpha < 1$ and $\gamma > 0$, and

$$\mu(n) = \begin{cases} \mu_{max} & \text{if } \mu'(n) > \mu_{max} \\ \mu_{min} & \text{if } \mu'(n) < \mu_{min} \\ \mu'(n) & \text{otherwise} \end{cases}$$

where $0 < \mu_{min} < \mu_{max}$. The initial step size parameter $\mu(0)$ is usually set equal to μ_{max}. The constant μ_{max} is selected to ensure the stability of the algorithm, and it is obtained from $\mu_{max} \leq \frac{2}{3\,\mathrm{tr}(R)}$, where tr(R) is the trace of the input correlation matrix R. And, μ_{min} is near the value of the fixed step size parameter in the conventional LMS algorithm. [7] recommends the typical values of α and γ as $\alpha = 0.97$ and $\gamma = 4.8 \times 10^{-4}$.

SIMULATION RESULT

In this section we give some examples to demonstrate the utility of our step size adjustment algorithm.

In eq. (1), we let $M = 4$. Thus, the signal $y(n)$ is generated by

$$y(n) = a_0 u(n) + a_1 u(n-1) + a_2 u(n-2) + a_3 u(n-3) + w(n) \qquad (16)$$

where $w(n)$ is zero mean white Gaussian noise with a variance of 1.0, 0.01, and 0.0001. Let $a_0 = 1.0, a_1 = 0.7$, $a_2 = 0.5$, and $a_3 = 0.2$. The values of p and q are selected according to eqs.(10) and (11). We initially used 0.5 for both, and 0.9 for d. We ran the LMS algorithm using eq.(13) to automatically adjust the step size. We compared the results with results obtained from using the conventional LMS algorithm with the best fixed step size obtained by trial and error, and the algorithms of Harris [5] and Kwong and Johnston [7].

As we see from the review above, in [5] the parameters m_0 and m_1 are important and they vary. So do the parameters α and γ in [7]. To achieve the best performance in both algorithms, we exhaustively searched for the most suitable values for the parameters. Our search began with the recommended values for those two algorithms. If the result was not achieved (i.e., no faster convergence rate than the conventional LMS or the steady state error was not lower), we then, by trial and error, used different values. For example, we searched for the most suitable values of m_0 and m_1 in [5] from $m_0 = m_1 = 2$ to $m_0 = m_1 - 1 = 4$. In [7] the recommended value of γ is 4.8×10^{-4}, and we tried values from 0.00048 to 0.48. So the results for [5] and [7] are the best we could find.

Example 1. The input is assumed to be an independent, zero mean, Gaussian random sequence with correlation matrix $\mathbf{R}_{uu} = \sigma^2 \mathbf{I}$. Here, we set $\sigma^2 = 1$; and the results are obtained from the average of 3000 trials of the algorithms. Fig.(1) shows results for $\sigma_w^2 = 1.0$, where (b) used $\mu = 0.1$; (c) used $\mu_{max} = 0.05$, $\mu_{min} = 5 \times 10^{-7}$, $m_0 = m_1 = 3$; (d) used $\mu_{max} = 0.1$, $\mu_{min} = 5 \times 10^{-7}$, $\alpha = 0.97$ and $\gamma = 0.48$. Fig.(2) shows results for $\sigma_w^2 = 0.01$, where (b) used $\mu = 0.1$; (c) used $\mu_{max} = 0.2$, $\mu_{min} = 5 \times 10^{-7}$, $m_0 = m_1 = 3$; (d) used $\mu_{max} = 0.1$, $\mu_{min} = 5 \times 10^{-7}$, $\alpha = 0.97$ and $\gamma = 0.48$. Fig.(3) shows results for $\sigma_w^2 = 0.0001$, where (b) used

$\mu = 0.1$; (c) used $\mu_{max} = 0.3$, $\mu_{min} = 5 \times 10^{-7}$, $m_0 = m_1 = 3$; (d) used $\mu_{max} = 0.25$, $\mu_{min} = 5 \times 10^{-7}$, $\alpha = 0.97$ and $\gamma = 0.48$. In all cases the algorithm proposed here performed better than the others.

Example 2. The input is a correlated sequence as in [7], where $u(n) = 0.9u(n-1) + v(n)$ and $v(n)$ is zero mean white Gaussian noise with a variance of unity. Fig.(4) shows results for $\sigma_w^2 = 1.0$, averaged over 1000 trials, where (b) used $\mu = 0.008$; (c) used $\mu_{max} = 0.01$, $\mu_{min} = 5 \times 10^{-7}$, $m_0 = m_1 - 1 = 4$; (d) used $\mu_{max} = 0.005$, $\mu_{min} = 5 \times 10^{-7}$, $\alpha = 0.97$ and $\gamma = 0.00048$. Fig.(5) shows results for $\sigma_w^2 = 0.01$, averaged over 500 trials, where (b) used $\mu = 0.008$; (c) used $\mu_{max} = 0.03$, $\mu_{min} = 1 \times 10^{-13}$, $m_0 = m_1 = 4$; (d) used $\mu_{max} = 0.007$, $\mu_{min} = 5 \times 10^{-7}$, $\alpha = 0.97$ and $\gamma = 0.48$. Fig.(6) shows results for $\sigma_w^2 = 0.0001$, averaged over 25 trials, where (b) used $\mu = 0.008$; (c) used $\mu_{max} = 0.03$, $\mu_{min} = 1 \times 10^{-13}$, $m_0 = m_1 = 4$; (d) used $\mu_{max} = 0.008$, $\mu_{min} = 5 \times 10^{-7}$, $\alpha = 0.97$ and $\gamma = 0.48$. Again, the proposed algorithm gave the best results.

REFERENCES

[1] B. Widrow and S.D. Sterns, Adaptive Signal Processing, Prentice Hall, Inc., 1985

[2] S. Haykin, Adaptive Filter Theory, Prentice Hall, Inc., 1985

[3] J.J. Shynk and S. Roy, "The LMS Algorithm with Momentum Updating",Proc. IEEE Int. Symp. Circuits Systems, June 1988

[4] S. Roy and J.J. Shynk, "Analysis of the Momentum LMS Algorithm", IEEE Trans. on Acoustics, Speech, and Signal Processing, Vol. 38, No. 12, 1990

[5] R.W. Harris, D.M. Chabries, and F.A. Bishop, "A Variable Step (VS) Adaptive Filter Algorithm", IEEE Trans. on Acoustics, Speech, and Signal Processing, Vol. 34, No. 2, April 1986

[6] T.J. Shan and T. Kailath, "Adaptive Algorithms With an Automatic Gain Control Feature", IEEE Trans. on Circuits and Systems, Vol. 35, No. 1, January 1988

[7] R.H.Kwong and E.W. Johnston, "A Variable Step Size LMS Algorithm", IEEE Trans. on Signal Processing, Vol. 40, No. 7, July 1992

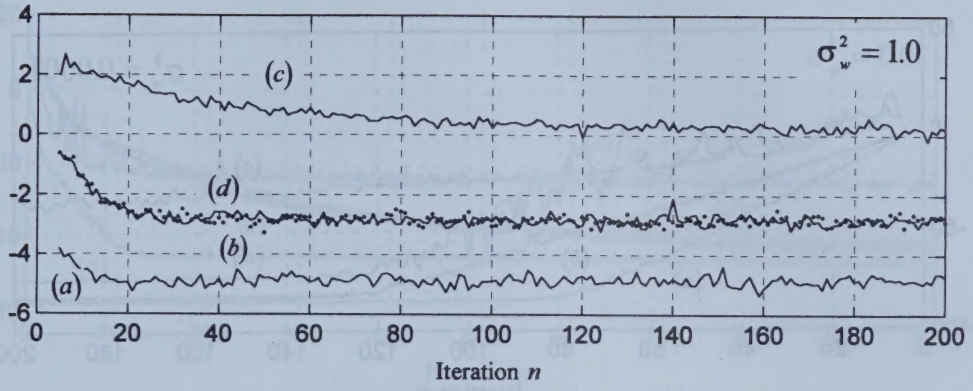


Fig.1 Plots of 10Log of mean squared error, the input with $\mathbf{R}_{uu} = \mathbf{I}$, where (a) is of our new algorithm, (b) of conventional LMS, (c) of Harris's, and (d) of Kwong's

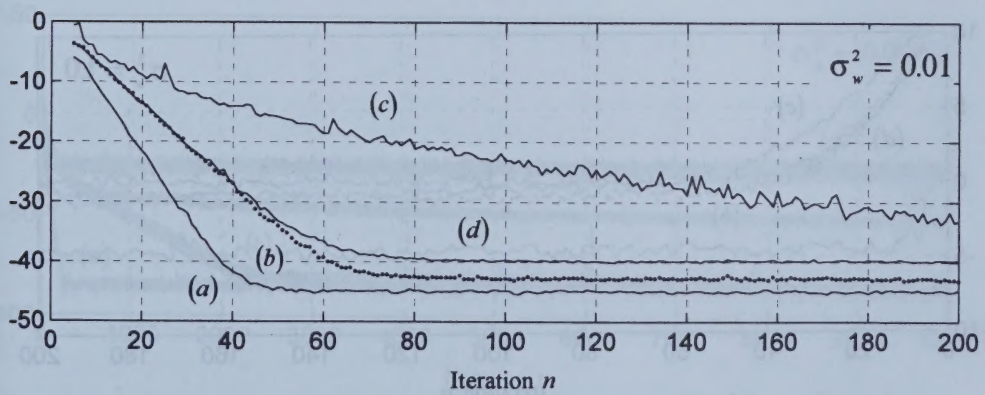


Fig.2 Plots of 10Log of mean squared error, the input with $\mathbf{R}_{uu} = \mathbf{I}$, where (a) is of our new algorithm, (b) of conventional LMS, (c) of Harris's, and (d) of Kwong's

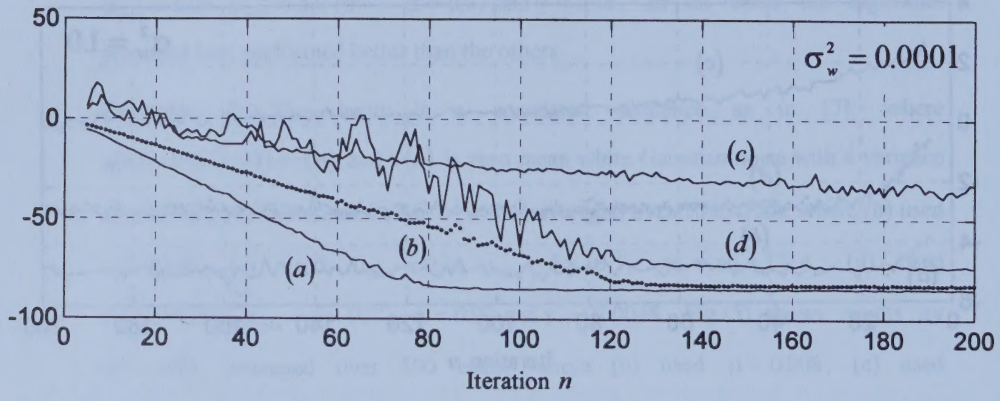


Fig.3 Plots of 10Log of mean squared error, the input with $\mathbf{R}_{uu} = \mathbf{I}$, where (a) is of our new algorithm, (b) of conventional LMS, (c) of Harris's, and (d) of Kwong's

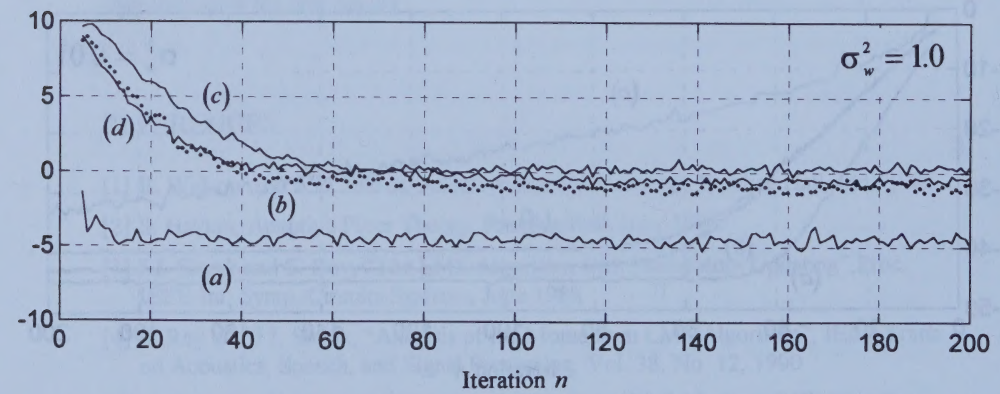


Fig.4 Plots of 10Log of mean squared error, the input is correlated, where (a) is of our new algorithm, (b) of conventional LMS, (c) of Harris's, and (d) of Kwong's

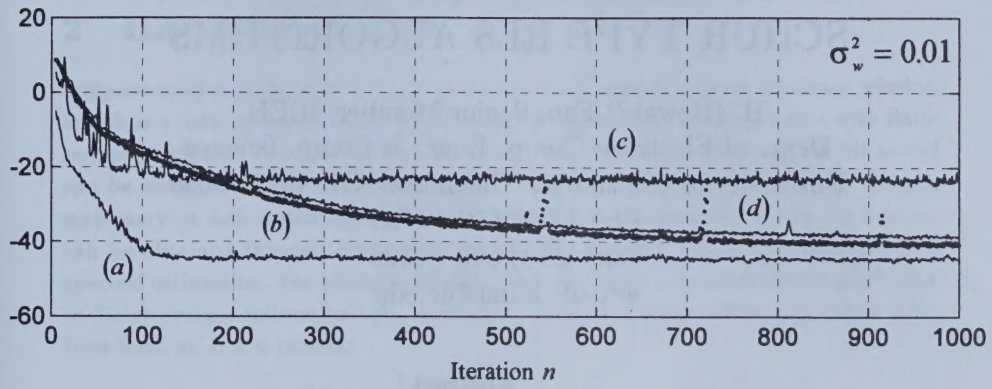


Fig.5 Plots of 10Log of mean squared error, the input is correlated, where (a) is of our new algorithm, (b) of conventional LMS, (c) of Harris's, and (d) of Kwong's

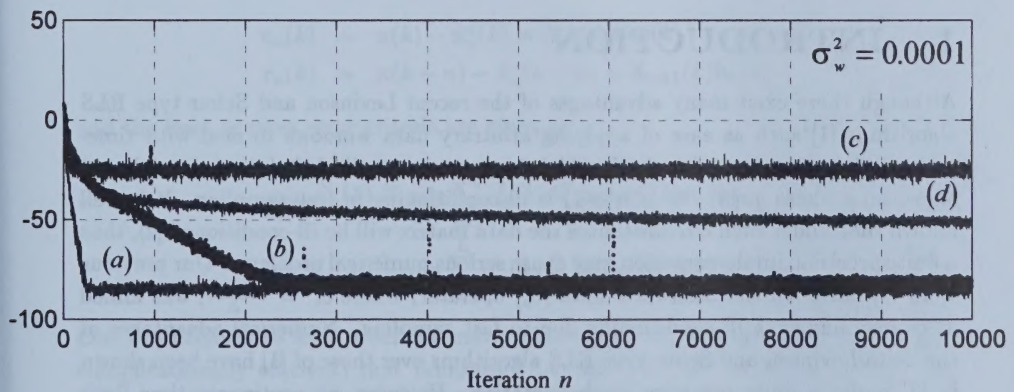


Fig.6 Plots of 10Log of mean squared error, the input is correlated, where (a) is of our new algorithm, (b) of conventional LMS, (c) of Harris's, and (d) of Kwong's

IMPROVED DELTA LEVINSON AND SCHUR TYPE RLS ALGORITHMS

H. (Howard) Fan, Senior Member, IEEE
Dept. of Electr. & Comp. Engr. & Comp. Science
University of Cincinnati, Cincinnati, OH 45221-0030
Ph.: (513) 556 - 4765
Fax: (513) 556 - 7326
e-mail: h.fan@uc.edu

Abstract [1]

In this paper previously published delta Levinson and Schur type RLS algorithms are revisited and their improved versions are proposed. It is seen that their improved versions converge, as the sampling interval vanishes, to time-varying continuous-time limit forms. Therefore numerical properties of the new versions proposed herein are further improved from the previous versions under fast sampling. Simulation examples confirm the expected results.

1 INTRODUCTION

Although there exist many advantages of the recent Levinson and Schur type RLS algorithms [1] such as ease of applying arbitrary data windows to deal with time-varying situations, one drawback which is common to all shift (q) operator based algorithms where $qx(n) = x(n+1)$ is ill-conditioning in fast sampling. It is well known that under such circumstances the data matrix will be ill-conditioned [2], thus a finite precision implementation may cause serious numerical problems. Our previous work [3] based on the backward delta (δ_b) operator, where $\delta_b = \frac{1-q^{-1}}{\Delta}$, was aimed at overcoming such ill-conditioning due to fast sampling. Numerical advantages of the delta Levinson and Schur type RLS algorithms over those of [1] have been shown in [3] under a finite precision implementation. However, no continuous-time limit forms were shown in [3]. In this paper, improved versions of [3] are developed. It will be shown that these new versions converge, as the sampling interval $\Delta \to 0$, to

[1] This work was supported by the Office of Naval Research under Grant N00014-96-1-0241.

their corresponding continuous-time limit forms. These limit forms are time-varying in nature, thus further revealing the ability of the algorithms of [1] and [3] in dealing with time-varying situations.

2 BACKGROUND

Suppose $\mathbf{x}(k) = [x(k), x(k-1), \cdots, x(k-L+1)]^T$ where L is an arbitrary window length is a data vector which is a sequence of samples of a random process with finite second moments. The process may or may not be stationary. Typically an AR model can be assumed on $x(k)$ with constant or time-varying coefficients corresponding to stationary or non-stationary processes. Least squares estimation of the parameters can be obtained through linear prediction. This gives an immediate application in spectral estimation. For adaptive filtering, one can easily extend the linear prediction to "joint process estimation" [4]. Thus only linear prediction will be considered here. Now form an $L \times n$ matrix:

$$\mathbf{X}_n(k) = [\mathbf{x}(k), \mathbf{x}(k-1), \cdots, \mathbf{x}(k-n+1)]$$

and define (in general) time-varying forward and backward prediction filter coefficients as

$$\mathbf{a}_n(k) = [1, a_{n1}(k), \cdots, a_{nn}(k)]^T, \quad \mathbf{b}_n(k) = [b_{nn}(k), \cdots, b_{n1}(k), 1]^T \tag{2.1}$$

When linear predictions are performed on the given data, the forward and backward prediction error *vectors* are defined by

$$\mathbf{e}_n(k) = \mathbf{x}(k) - \hat{\mathbf{x}}_n^e(k) = \mathbf{X}_{n+1}(k)\mathbf{a}_n(k) \tag{2.2}$$

$$\mathbf{r}_n(k) = \mathbf{x}(k-n) - \hat{\mathbf{x}}_n^r(k-n) = \mathbf{X}_{n+1}(k)\mathbf{b}_n(k) \tag{2.3}$$

and the least squares error criteria are given as

$$\min_{\mathbf{a}_n(k)} E_n(k) \triangleq \min_{\mathbf{a}_n(k)} \sum_{i=1}^{L} e_{ni}^2(k) \triangleq \min_{\mathbf{a}_n(k)} \mathbf{e}_n^T(k)\mathbf{e}_n(k) \tag{2.4}$$

$$\min_{\mathbf{b}_n(k)} R_n(k) \triangleq \min_{\mathbf{b}_n(k)} \sum_{i=1}^{L} r_{ni}^2(k) \triangleq \min_{\mathbf{b}_n(k)} \mathbf{r}_n^T(k)\mathbf{r}_n(k). \tag{2.5}$$

Our objective is to find order recursive relations for $\mathbf{a}_n(k)$ and $\mathbf{b}_n(k)$ (and hence computationally efficient) that minimize (2.4) and (2.5) for an increasing k. The Levinson type RLS algorithm of [1] fulfills this objective and gives,

$$\mathbf{a}_n(k) = \begin{bmatrix} \mathbf{a}_{n-1}(k) \\ 0 \end{bmatrix} + \Gamma_n^e(k) \begin{bmatrix} 0 \\ \mathbf{b}_{n-1}(k-1) \end{bmatrix} \tag{2.6}$$

$$\mathbf{b}_n(k) = \begin{bmatrix} 0 \\ \mathbf{b}_{n-1}(k-1) \end{bmatrix} + \Gamma_n^r(k) \begin{bmatrix} \mathbf{a}_{n-1}(k) \\ 0 \end{bmatrix} \tag{2.7}$$

where

$$\Gamma_n^e(k) = -C_{n-1}(k)/R_{n-1}(k-1), \quad \Gamma_n^r(k) = -C_{n-1}(k)/E_{n-1}(k) \tag{2.8}$$

$$C_n(k) = \mathbf{x}^T(k)\mathbf{X}_{n+1}(k-1)\mathbf{b}_n(k-1) \quad (2.9)$$
$$E_n(k) = E_{n-1}(k) + \Gamma_n^e(k)C_{n-1}(k) \quad (2.10)$$
$$R_n(k) = R_{n-1}(k-1) + \Gamma_n^r(k)C_{n-1}(k) \quad (2.11)$$

Since (2.9) requires inner product computation, the Levinson type RLS algorithm is not amenable to parallel computation. On the other hand, the Schur type RLS algorithm overcomes this difficulty, and can be obtained by defining Schur variables as, for $j \geq n+1$,

$$C_{n,j}^e(k) = \mathbf{x}^T(k-j)\mathbf{e}_n(k) = \mathbf{x}^T(k-j)\mathbf{X}_{n+1}(k)\mathbf{a}_n(k) \quad (2.12)$$
$$C_{n,j}^r(k) = \mathbf{x}^T(k-j)\mathbf{r}_n(k) = \mathbf{x}^T(k-j)\mathbf{X}_{n+1}(k)\mathbf{b}_n(k) \quad (2.13)$$

and left multiplying $\mathbf{x}^T(k-j)\mathbf{X}_{n+1}(k)$ to (2.6) and (2.7),

$$C_{n,j}^e(k) = C_{n-1,j}^e(k) + \Gamma_n^e(k)C_{n-1,j-1}^r(k-1) \quad (2.14)$$
$$C_{n,j}^r(k) = C_{n-1,j-1}^r(k-1) + \Gamma_n^r(k)C_{n-1,j}^e(k),\ j = n+1,\cdots,N. \quad (2.15)$$

Setting to zero the derivatives of (2.4) and (2.5) one obtains the normal equations,

$$\mathbf{X}_n^T(k-1)\mathbf{X}_{n+1}(k)\mathbf{a}_n(k) = \mathbf{0} \quad (2.16)$$
$$\mathbf{X}_n^T(k-1)\mathbf{X}_{n+1}(k-1)\mathbf{b}_n(k-1) = \mathbf{0} \quad (2.17)$$

where $\mathbf{0}$ denotes a zero vector. Therefore, $C_n(k)$ in (2.9) can be calculated as

$$\begin{aligned} C_n(k) &= \mathbf{x}^T(k)\mathbf{X}_{n+1}(k-1)\mathbf{b}_n(k-1) \\ &+ [a_{n1}(k), \cdots, a_{nn}(k)]\mathbf{X}_n^T(k-1)\mathbf{X}_{n+1}(k-1)\mathbf{b}_n(k-1) \\ &= \mathbf{a}_n^T(k)\mathbf{X}_{n+1}^T(k)\mathbf{X}_{n+1}(k-1)\mathbf{b}_n(k-1) \\ &= \mathbf{a}_n^T(k)\mathbf{X}_{n+1}^T(k)\mathbf{X}_n(k-1)\begin{bmatrix} b_{nn}(k-1) \\ \vdots \\ b_{n1}(k-1) \end{bmatrix} + \mathbf{a}_n^T(k)\mathbf{X}_{n+1}^T(k)\mathbf{x}(k-n-1) \\ &= \mathbf{x}^T(k-n-1)\mathbf{X}_{n+1}(k)\mathbf{a}_n(k) = C_{n,n+1}^e(k) \end{aligned} \quad (2.18)$$

(2.14) and (2.15) are the key equations in the q-domain Schur type RLS algorithm. See [1] for details on parallel implementation.

3 δ_b-LEVINSON RLS ALGORITHM AND ITS LIMIT FORM

Define a δ_b-domain data matrix as follows:

$$\mathbf{Z}_n(k) = \left[\mathbf{x}(k), \delta_b\mathbf{x}(k), \cdots, \delta_b^{n-1}\mathbf{x}(k)\right]. \quad (3.1)$$

Then it is easily seen that the following relationship exists:

$$\mathbf{Z}_n(k) = \mathbf{X}_n(k)T_{n-1} \quad (3.2)$$

where T_n is an $(n+1)\times(n+1)$ upper triangular matrix whose entries are [3]

$$(T_n)_{l,j} = \frac{(-1)^l}{\Delta^j}\binom{j}{l}, \quad 0 \le l, j \le n, \quad l \le j. \tag{3.3}$$

Also define

$$\boldsymbol{\alpha}_n(k) \triangleq [\alpha_{n0}(k), \alpha_{n1}(k), \cdots, \alpha_{nn}(k)]^T \triangleq \Delta^{-n} T_n^{-1} \mathbf{a}_n(k) \tag{3.4}$$

$$\boldsymbol{\beta}_n(k) \triangleq [\beta_{nn}(k), \cdots, \beta_{n1}(k), \beta_{n0}(k)]^T \triangleq \Delta^{-n} T_n^{-1} \mathbf{b}_n(k) \tag{3.5}$$

Note that $\beta_{n0}(k) = (-1)^n$. Use this transformation, the δ_b-domain prediction errors become

$$\tilde{\mathbf{e}}_n(k) = \mathbf{Z}_{n+1}(k)\boldsymbol{\alpha}_n(k) = \Delta^{-n}\mathbf{e}_n(k) \tag{3.6}$$

$$\tilde{\mathbf{r}}_n(k) = \mathbf{Z}_{n+1}(k)\boldsymbol{\beta}_n(k) = \Delta^{-n}\mathbf{r}_n(k) \tag{3.7}$$

and the δ_b-domain least square error criteria become

$$\min_{\boldsymbol{\alpha}_n(k)} \tilde{E}_n(k) = \min_{\boldsymbol{\alpha}_n(k)} \tilde{\mathbf{e}}_n^T(k)\tilde{\mathbf{e}}_n(k), \quad \min_{\boldsymbol{\beta}_n(k)} \tilde{R}_n(k) = \min_{\boldsymbol{\beta}_n(k)} \tilde{\mathbf{r}}_n^T(k)\tilde{\mathbf{r}}_n(k) \tag{3.8}$$

Instead of performing the new minimization (3.8), the δ_b-Levinson type RLS algorithm can be simply obtained by linearly transforming the q-domain one. Left multiplying $\Delta^{-n}T_n^{-1}$ to (2.6) and (2.7) yields

$$\boldsymbol{\alpha}_n(k) = \frac{1}{\Delta}\left\{\begin{bmatrix}\boldsymbol{\alpha}_{n-1}(k)\\ 0\end{bmatrix} + \Gamma_n^e(k)\left(\begin{bmatrix}\boldsymbol{\beta}_{n-1}(k-1)\\ 0\end{bmatrix} - \Delta\begin{bmatrix}0\\ \boldsymbol{\beta}_{n-1}(k-1)\end{bmatrix}\right)\right\} \tag{3.9}$$

$$\boldsymbol{\beta}_n(k) = \frac{1}{\Delta}\left\{\begin{bmatrix}\boldsymbol{\beta}_{n-1}(k-1)\\ 0\end{bmatrix} - \Delta\begin{bmatrix}0\\ \boldsymbol{\beta}_{n-1}(k-1)\end{bmatrix} + \Gamma_n^r(k)\begin{bmatrix}\boldsymbol{\alpha}_{n-1}(k)\\ 0\end{bmatrix}\right\} \tag{3.10}$$

This "two term recursion" version has no limit as $\Delta \to 0$. To derive a "three term recursion" which has a limit as $\Delta \to 0$, one notes from (3.8) the scaled versions of $E_n(k)$ and $R_n(k)$, and defines that of $C_n(k)$, similar to [5],

$$\tilde{E}_n(k) = \Delta^{-2n}E_n(k), \ \tilde{R}_n(k) = \Delta^{-2n}R_n(k), \ \tilde{C}_n(k) = \Delta^{-n}C_n(k) \tag{3.11}$$

Then, from (2.8)-(2.11) it is easily seen that

$$\begin{aligned} E_n(k) &= E_{n-1}(k)[1 - \Gamma_n^e(k)\Gamma_n^r(k)] \\ R_n(k) &= R_{n-1}(k-1)[1 - \Gamma_n^e(k)\Gamma_n^r(k)] \end{aligned} \tag{3.12}$$

Therefore, one obtains

$$\frac{\tilde{E}_n(k)}{\tilde{E}_{n-1}(k)} = \frac{\tilde{R}_n(k)}{\tilde{R}_{n-1}(k-1)} = \frac{1 - \Gamma_n^e(k)\Gamma_n^r(k)}{\Delta^2} \tag{3.13}$$

Furthermore, from (2.8)-(2.11) and using (3.11), it can be easily seen that

$$\frac{\Gamma_{n+1}^r(k)}{\Gamma_n^r(k)} = \frac{E_{n-1}(k)C_n(k)}{E_n(k)C_{n-1}(k)} = \frac{\tilde{R}_{n-1}(k-1)\tilde{C}_n(k)}{\Delta\tilde{R}_n(k)\tilde{C}_{n-1}(k)} \tag{3.14}$$

Now manipulate (3.10) to express $\boldsymbol{\alpha}_{n-1}(k)$ and $\boldsymbol{\alpha}_n(k)$ in terms of $\boldsymbol{\beta}_i(k)$, $i = n-1, n, n+1$ and then substitute into (3.9). The following results

$$\begin{aligned}
\boldsymbol{\beta}_{n+1}(k) &= \frac{1}{\Delta}\begin{bmatrix}\boldsymbol{\beta}_n(k-1)\\ 0\end{bmatrix} - \begin{bmatrix}0\\ \boldsymbol{\beta}_n(k-1)\end{bmatrix} + \frac{\Gamma^r_{n+1}(k)}{\Delta\Gamma^r_n(k)}\begin{bmatrix}\boldsymbol{\beta}_n(k)\\ 0\end{bmatrix} \\
&+ \frac{\Gamma^r_{n+1}(k)}{\Delta^2}[\Gamma^e_n(k) - \frac{1}{\Gamma^r_n(k)}]\begin{bmatrix}\boldsymbol{\beta}_{n-1}(k-1)\\ 0\\ 0\end{bmatrix} \\
&- \frac{\Gamma^r_{n+1}(k)}{\Delta}[\Gamma^e_n(k) - \frac{1}{\Gamma^r_n(k)}]\begin{bmatrix}0\\ \boldsymbol{\beta}_{n-1}(k-1)\\ 0\end{bmatrix} \\
&= \frac{1}{\Delta}\left\{\begin{bmatrix}\boldsymbol{\beta}_n(k-1)\\ 0\end{bmatrix} + \frac{\tilde{R}_{n-1}(k-1)\tilde{C}_n(k)}{\Delta\tilde{R}_n(k)\tilde{C}_{n-1}(k)}\begin{bmatrix}\boldsymbol{\beta}_n(k)\\ 0\end{bmatrix}\right\} - \begin{bmatrix}0\\ \boldsymbol{\beta}_n(k-1)\end{bmatrix} \\
&- \frac{\tilde{C}_n(k)}{\Delta\tilde{C}_{n-1}(k)}\begin{bmatrix}\boldsymbol{\beta}_{n-1}(k-1)\\ 0\\ 0\end{bmatrix} + \frac{\tilde{C}_n(k)}{\tilde{C}_{n-1}(k)}\begin{bmatrix}0\\ \boldsymbol{\beta}_{n-1}(k-1)\\ 0\end{bmatrix} \qquad (3.15)
\end{aligned}$$

where (3.13) and (3.14) were used in the last equality. What remains now is the calculation of $\tilde{R}_n(k)$ and $\tilde{C}_n(k)$. From (2.9) and (3.11), using (3.2)-(3.5),

$$\begin{aligned}
\tilde{C}_n(k) &= \Delta^{-n}C_n(k) = \mathbf{x}^T(k)\mathbf{X}_{n+1}(k-1)[\Delta^{-n}\mathbf{b}_n(k-1)] \\
&= \mathbf{x}^T(k)\mathbf{Z}_{n+1}(k-1)\boldsymbol{\beta}_n(k-1). \qquad (3.16)
\end{aligned}$$

From the definition of $R_n(k)$ in (2.5) and (2.3), and using the normal equation (2.17), it is seen that

$$\begin{aligned}
R_n(k) &= \mathbf{r}^T_n(k)\mathbf{r}_n(k) = \mathbf{b}^T_n(k)\mathbf{X}^T_{n+1}(k)\mathbf{X}_{n+1}(k)\mathbf{b}_n(k) \\
&= \mathbf{x}^T(k-n)\mathbf{X}_{n+1}(k)\mathbf{b}_n(k) \qquad (3.17)
\end{aligned}$$

Combining (3.17) with (2.17) gives the augmented normal equation

$$\mathbf{X}^T_{n+1}(k)\mathbf{X}_{n+1}(k)\mathbf{b}_n(k) = \begin{bmatrix}\mathbf{0}\\ R_n(k)\end{bmatrix}. \qquad (3.18)$$

Now transform (3.18) into the δ_b-domain by left multiplying $\Delta^{-n}T^T_n$.

$$T^T_n\mathbf{X}^T_{n+1}(k)\mathbf{X}_{n+1}(k)T_n[\Delta^{-n}T^{-1}_n\mathbf{b}_n(k)] = T^T_n\begin{bmatrix}\mathbf{0}\\ \frac{R_n(k)}{\Delta^n}\end{bmatrix}. \qquad (3.19)$$

This gives

$$\mathbf{Z}^T_{n+1}(k)\mathbf{Z}_{n+1}(k)\boldsymbol{\beta}_n(k) = (-1)^n\begin{bmatrix}\mathbf{0}\\ \tilde{R}_n(k)\end{bmatrix}. \qquad (3.20)$$

The last entry of (3.20) yields

$$\tilde{R}_n(k) = (-1)^n[\delta_b{}^n\mathbf{x}^T(k)]\mathbf{Z}_{n+1}(k)\boldsymbol{\beta}_n(k). \qquad (3.21)$$

(3.15),(3.16) and (3.21) constitute the new δ_b-Levinson type RLS algorithm. It is similar in form to [5], but is capable of dealing with given data in a time-varying situation.

This aspect is especially clear from its limit form as $\Delta \to 0$. To see this, note that $\Gamma_n^r(k) \overset{\Delta \to 0}{\to} (-1)^{n+1}$ [5]. Then using (3.14) in (3.15) and letting $k\Delta \overset{\Delta \to 0}{\to} t$, one obtains

$$\boldsymbol{\beta}_{n+1}(t) = -\begin{bmatrix} \dot{\boldsymbol{\beta}}_n(t) \\ 0 \end{bmatrix} - \begin{bmatrix} 0 \\ \boldsymbol{\beta}_n(t) \end{bmatrix} + \frac{\tilde{R}_n(t)}{\tilde{R}_{n-1}(t)} \begin{bmatrix} \boldsymbol{\beta}_{n-1}(t) \\ 0 \\ 0 \end{bmatrix} \tag{3.22}$$

$$\tilde{R}_n(t) = (-1)^n [\mathbf{x}^{(n)}(t)]^T \mathbf{Z}_{n+1}(t) \boldsymbol{\beta}_n(t) \tag{3.23}$$

It can be shown that $\tilde{R}_n(t)$ is the n-th order "prediction" error energy. This is seen by taking the limit of (3.7),

$$\tilde{\mathbf{r}}(t) = \mathbf{Z}_{n+1}(t) \boldsymbol{\beta}_n(t) = \left[\mathbf{x}(t), \dot{\mathbf{x}}(t), \cdots, \mathbf{x}^{(n)}(t)\right] \boldsymbol{\beta}_n(t) \tag{3.24}$$

Minimizing $\tilde{R}_n(t)$ as a limit of (3.8) with respect to $\beta_{n1}(k), \cdots, \beta_{nn}(k)$ yields the continuous-time normal equation

$$\mathbf{Z}_n^T(t) \mathbf{Z}_{n+1}(t) \boldsymbol{\beta}_n(t) = \mathbf{0} \tag{3.25}$$

Substituting (3.25) back into $\tilde{R}_n(t) = \tilde{\mathbf{r}}^T(t)\tilde{\mathbf{r}}(t)$ yields (3.23). Therefore $\tilde{\mathbf{r}}(t)$ represents the error in "predicting" $\mathbf{x}^{(n)}(t)$ based on $\mathbf{x}(t), \dot{\mathbf{x}}(t), \cdots, \mathbf{x}^{(n-1)}(t)$. (3.22) is the time-varying version of the Pham-LeBreton recursions [6]. Due to the existence of this limit, for Δ small the new δ_b-Levinson type RLS algorithm would have much better numerical properties than the q-version.

4 δ_b-SCHUR RLS ALGORITHM AND ITS LIMIT FORM

Define vectors consisting the elements of (2.12) and (2.13) as

$$\mathbf{C}_{n,N}^e(k) = \left[C_{n,n}^e(k), C_{n,n+1}^e(k), \cdots, C_{n,N}^e(k)\right]^T \tag{4.1}$$

$$\mathbf{C}_{n,N}^r(k) = \left[C_{n,n}^r(k), C_{n,n+1}^r(t), \cdots, C_{n,N}^r(k)\right]^T \tag{4.2}$$

Note that these elements are those on the left hand sides (LHS) of (2.14) and (2.15) with j extended to include n. This extension is needed in developing the δ_b-Schur type RLS algorithm, and will become apparent in (4.9). As n increases, the length of the vectors (4.1) and (4.2) decreases whether $j = n$ (for every n) is included or not. For example, at every n after (2.14) and (2.15) are executed, the $j = n + 1$ element on the LHS of (2.14) is taken as $C_n(k)$ and then discarded together with the $j = N$ element on the LHS of (2.15) [1]. This is because these elements are no longer needed in executing (2.14) and (2.15) for the next stage with an incremented value of n. The same situation is true when $j = n$ is included in (2.14) and (2.15). With the definition of (4.1) and (4.2), however, this order decrease is accounted for now since

(2.14) and (2.15) with j extended to include n can now be written as

$$\begin{bmatrix} \mathbf{C}^e_{n,N}(k) \\ \mathbf{C}^r_{n,N}(k) \end{bmatrix} = \begin{bmatrix} I & \Gamma^e_n(k)I \\ \Gamma^r_n(k)I & I \end{bmatrix} \begin{bmatrix} [\mathbf{0}\ I] & 0 \\ 0 & [I\ \mathbf{0}] \end{bmatrix} \bullet \begin{bmatrix} \mathbf{C}^e_{n-1,N}(k) \\ \mathbf{C}^r_{n-1,N}(k-1) \end{bmatrix} \tag{4.3}$$

Define the δ_b-Schur variables:

$$\begin{bmatrix} \tilde{\mathbf{C}}^e_{n,N}(k) \\ \tilde{\mathbf{C}}^r_{n,N}(k) \end{bmatrix} = \Delta^{-n} \begin{bmatrix} T^T_{N-n} & \mathbf{O} \\ \mathbf{O} & T^T_{N-n} \end{bmatrix} \begin{bmatrix} \mathbf{C}^e_{n,N}(k) \\ \mathbf{C}^r_{n,N}(k) \end{bmatrix} \tag{4.4}$$

i.e., element-wise,

$$\begin{aligned} \tilde{C}^e_{n,j}(k) &= \delta_b{}^{j-n}\mathbf{x}^T(k-n)\mathbf{Z}_{n+1}(k)\boldsymbol{\alpha}_n(k) \\ \tilde{C}^r_{n,j}(k) &= \delta_b{}^{j-n}\mathbf{x}^T(k-n)\mathbf{Z}_{n+1}(k)\boldsymbol{\beta}_n(k),\ j = n,\cdots,N. \end{aligned} \tag{4.5}$$

Now left multiply $\Delta^{-n}\begin{bmatrix} T^T_{N-n} & \mathbf{O} \\ \mathbf{O} & T^T_{N-n} \end{bmatrix}$ to (4.3) and go through similar calculations as [3] to obtain,

$$\begin{bmatrix} \tilde{\mathbf{C}}^e_{n,N}(k) \\ \tilde{\mathbf{C}}^r_{n,N}(k) \end{bmatrix} = \Delta^{-1} \begin{bmatrix} I & \Gamma^e_n(k)I \\ \Gamma^r_n(k)I & I \end{bmatrix} \begin{bmatrix} [I\ \mathbf{0}] - \Delta\,[\mathbf{0}\ I] & \mathbf{O} \\ \mathbf{O} & [I\ \mathbf{0}] \end{bmatrix} \bullet \begin{bmatrix} \tilde{\mathbf{C}}^e_{n-1,N}(k) \\ \tilde{\mathbf{C}}^r_{n-1,N}(k-1) \end{bmatrix}. \tag{4.6}$$

Or element-wise,

$$\begin{aligned} \tilde{C}^e_{n,j}(k) &= \Delta^{-1}\tilde{C}^e_{n-1,j-1}(k) - \tilde{C}^e_{n-1,j}(k) + \Delta^{-1}\Gamma^e_n(k)\tilde{C}^r_{n-1,j-1}(k-1) \\ \tilde{C}^r_{n,j}(k) &= \Delta^{-1}\Gamma^r_n(k)\tilde{C}^e_{n-1,j-1}(k) - \Gamma^r_n(k)\tilde{C}^e_{n-1,j}(k) + \Delta^{-1}\tilde{C}^r_{n-1,j-1}(k-1), \\ & \quad j = n,\cdots,N \end{aligned} \tag{4.7}$$

By substituting one into the other of (4.7) and using (3.13) and (3.14), in a way similar to the δ_b-Levinson case, the following "three-term" recursion results,

$$\begin{aligned} \tilde{C}^e_{n+1,j+1}(k+1) &= \frac{1}{\Delta}\{\tilde{C}^e_{n,j}(k+1) + \frac{\tilde{R}_{n-1}(k-1)\tilde{C}_n(k+1)}{\Delta\tilde{R}_n(k)\tilde{C}_{n-1}(k)}\tilde{C}^e_{n,j}(k)\} - \tilde{C}^e_{n,j+1}(k+1) \\ &\quad - \frac{\tilde{C}_n(k+1)}{\Delta\tilde{C}_{n-1}(k)}\tilde{C}^e_{n-1,j-1}(k) + \frac{\tilde{C}_n(k+1)}{\tilde{C}_{n-1}(k)}\tilde{C}^e_{n-1,j}(k). \end{aligned} \tag{4.8}$$

To calculate $\tilde{C}_n(k)$, use (2.18), (2.12) and (4.5) to obtain

$$\begin{aligned} \tilde{C}_n(k) &= \Delta^{-n}C_n(k) = \mathbf{x}^T(k-n-1)\mathbf{X}_{n+1}(k)[\Delta^{-n}\mathbf{a}_n(k)] \\ &= \mathbf{x}^T(k-n-1)\mathbf{Z}_{n+1}(k)\boldsymbol{\alpha}_n(k) \\ &= [\mathbf{x}^T(k-n) - \Delta\delta_b\mathbf{x}^T(k-n)]\mathbf{Z}_{n+1}(k)\boldsymbol{\alpha}_n(k) \\ &= \tilde{C}^e_{n,n}(k) - \Delta\tilde{C}^e_{n,n+1}(k). \end{aligned} \tag{4.9}$$

To calculate $\tilde{R}_n(k)$, however, one can not directly use the data as in the δ_b-Levinson case since (3.21) involves an inner product. One can divide (2.10) and (2.11) by Δ^{2n} to obtain

$$\tilde{E}_n(k) = \frac{\tilde{E}_{n-1}(k)}{\Delta^2} - \frac{\tilde{C}^2_{n-1}(k)}{\Delta^{2n}\tilde{R}_{n-1}(k-1)}$$

$$\tilde{R}_n(k) = \frac{\tilde{R}_{n-1}(k-1)}{\Delta^2} - \frac{\tilde{C}^2_{n-1}(k)}{\Delta^{2n}\tilde{E}_{n-1}(k)} \tag{4.10}$$

The limit of (4.8) as $\Delta \to 0$ is easily obtained which is very similar to the δ_b-Levinson case.

$$\tilde{C}^e_{n+1,j+1}(t) = \frac{d}{dt}\tilde{C}^e_{n,j}(t) - \tilde{C}^e_{n,j+1}(t) + \frac{\tilde{R}_n(t)}{\tilde{R}_{n-1}(t)}\tilde{C}^e_{n-1,j-1}(t). \tag{4.11}$$

And the limit of (4.10) can theoretically be obtained by observing

$$\begin{aligned}\tilde{R}_n(k) &= \tilde{R}_{n-1}(k-1)\frac{1}{\Delta^2}[1 - \frac{\tilde{C}^2_{n-1}(k)}{\Delta^{2n-2}\tilde{E}_{n-1}(k)\tilde{R}_{n-1}(k-1)}] \\ &= \tilde{R}_{n-1}(k-1)[\frac{1-\Gamma^e_n(k)\Gamma^r_n(k)}{\Delta^2}]\end{aligned} \tag{4.12}$$

where (3.11) and (2.8) are used. The factor in the square brackets is the same as (3.13) and has a limit as $\Delta \to 0$ [3], [5]. Thus the limit of $\tilde{R}_n(k)$ exists. However, since the data is not directly used in (4.10), the numerical properties of the δ_b-Schur RLS algorithm will not be as good as those of the δ_b-Levinson under a finite precision implementation. In addition, for very small Δ the divisions in (4.10) may lose numerical accuracy. This is the price one needs to pay in exchange for parallel processing which would be similar to [3] but is not discussed here. One can define $\bar{E}_n(k) = \Delta^{-n}E_n(k)$ and $\bar{R}_n(k) = \Delta^{-n}R_n(k)$ to slightly improve the situation. Nevertheless, improvement over the q-Schur version can still be substantial.

5 SIMULATION RESULTS

To compare with the results of [3], the exact same setting as that of [3] is used here. That is, an AR(4) model is used to generate the data $\{x(k)\}$, whose double poles correspond to continuous-time poles $-3 \pm j3$ sampled with $T = \Delta = 0.01$ and 0.004, respectively. The resulting discrete-time double poles are at $0.97044e^{\pm j1.7188°}$ and $0.98807e^{\pm j0.6875°}$, respectively. Prewindowing with a forgetting factor λ is used. The double precision provided by a Sun SPARCstation 10 is considered as infinite precision, and the relative error energy is defined as

$$Re(k) = \frac{|\tilde{R}^{fp}_4(k) - \tilde{R}^{\infty}_4(k)|}{\tilde{R}^{\infty}_4(k)} \tag{5.1}$$

where the superscript fp means finite precision, and ∞ for infinite precision. What is plotted is $\bar{Re}(k)$, ensemble averages of 30 runs of $Re(k)$, for a given number of binary bits used in the finite precision implementation, and the time averaged versions of $\bar{Re}(k)$ over the interval $k = 151$ to 400, $\bar{\bar{Re}}$, for various numbers of binary bits. The improvement over the results of [3] is clear.

6 Reference

[1] P. Strobach, "New forms of Levinson and Schur algorithms," IEEE SP Magazine, vol. 8, no. 1, pp.12-36, Jan. 1991.

[2] H.Fan,"A structural view of asymptotic convergence speed of adaptive IIR filtering algorithms - Part I : Infinite precision implementation,"*IEEE Trans. on Signal Processing,* vol. 41, no. 4, pp. 1493-1517, April 1993.

[3] H. Fan and X.Liu, "Delta Levinson and Schur-type RLS algorithms for adaptive signal processing," *IEEE Trans. on Signal Processing,* vol. 42, no. 7, pp. 1629-1639, June 1994.

[4] S. Haykin, *Adaptive Filter Theory.* 2nd ed., Englewood Cliffs, NJ:Prentice-Hall, 1991.

[5] R. Vijayan, H. V. Poor, J. B. Moore, and G. C. Goodwin, "A Levinson-type algorithm for modeling fast-sampled data," *IEEE Trans. on Automatic Control,* vol. AC-36, no. 3, pp. 314-321, March 1991.

[6] D. T. Pham and A. Le Breton, "Levinson-Durbin type algorithms for continuous-time autoregressive models and applications," *Math. Control, Signals and Systems,* vol. 4, pp. 69-79, 1991.

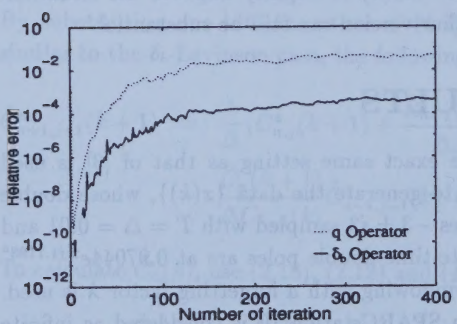


Figure 1(a): δ_b-Levinson RLS, ensemble averaged relative errors $\bar{Re}(k)$ versus k. $T = 0.01$,$\lambda = 0.99$, 45 bits.

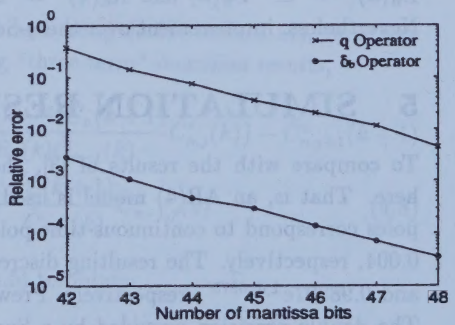


Figure 1(b):Time averaged relative errors $\bar{\bar{Re}}$ of Figure 1(a) versus number of binary bits.

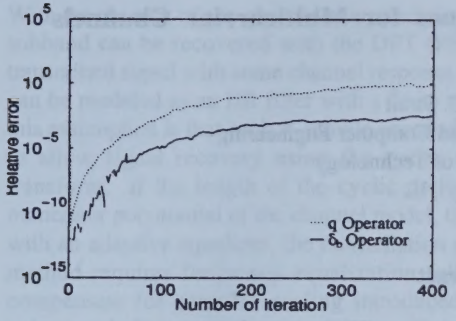


Figure 2(a): δ_b-Levinson RLS, ensemble averaged relative errors $\bar{R}e(k)$ versus k. $T = 0.004$,$\lambda = 0.99$, 50 bits.

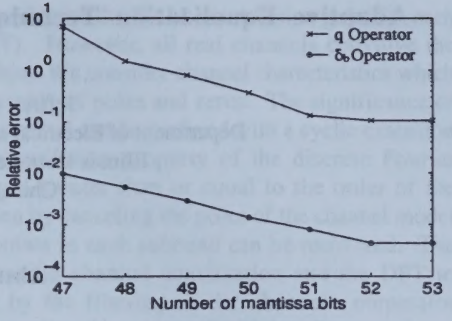


Figure 2(b): Time averaged relative errors $\bar{\bar{R}}e$ of Figure 2(a) versus number of binary bits.

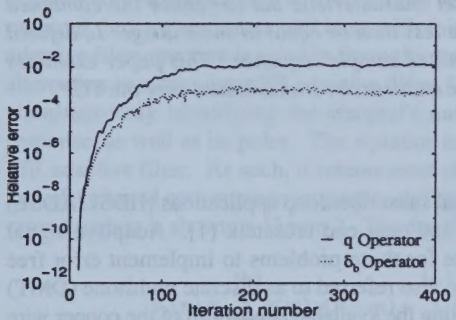


Figure 3(a): δ_b-Schur RLS, ensemble averaged relative errors $\bar{R}e(k)$ versus k. $T = 0.01$,$\lambda = 0.99$, 45 bits.

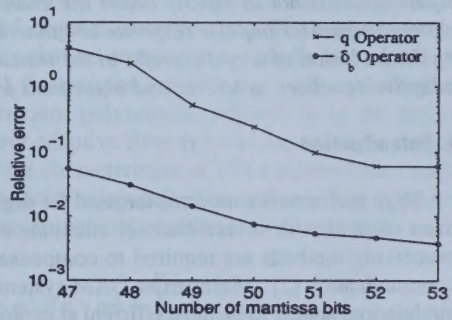


Figure 3(b): Time averaged relative errors $\bar{\bar{R}}e$ of Figure 3(a) versus number of binary bits.

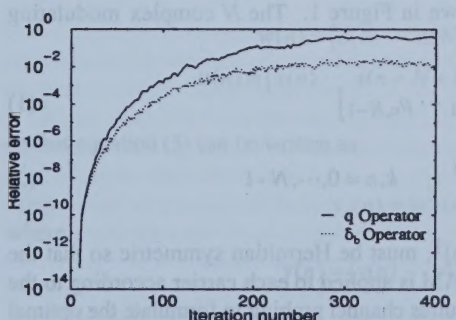


Figure 4(a):δ_b-Schur RLS, ensemble averaged relative errors $\bar{R}e(k)$ versus k. $T = 0.004$,$\lambda = 0.99$, 50 bits.

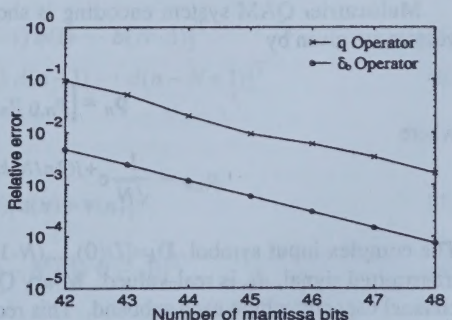


Figure 4(b): Time averaged relative errors $\bar{\bar{R}}e$ of Figure 4(a) versus number of binary bits.

Adaptive Equalization Techniques for Multicarrier Channels

Jeffrey C. Strait
Department of Electrical and Computer Engineering
Illinois Institute of Technology
Chicago, IL

Abstract

Communications bandwidth requirements for emerging digital applications far exceed the capabilities of the existing telecommunications transmission path. Multicarrier quadrature amplitude modulation systems offer an efficient method of maximizing the transmission bitrate over the twisted copper pair (TCP). However, the harsh attenuation and crosstalk characteristics of the TCP require careful equalization to reduce the achievable bit error rate. Because this modulation technique relies on the fast Fourier transform, the goal of equalization is not to exactly invert the channel characteristic but to reduce the combined channel-equalizer impulse response length to be less than or equal to some integer L, defined by the inclusion of a cyclic prefix in the transmitted sample sequence. This paper examines adaptive equalizer structures and algorithms for multicarrier transmission over the TCP.

1. Introduction

High performance modems targeted for digital subscriber loop applications (HDSL/ADSL) must content with severe channel attenuation and near-end crosstalk [1]. Adaptive signal processing methods are required to compensate for these problems to implement error free communication [2]. Multicarrier QAM systems, also referred to as discrete multitone (DMT) modulation, appear to be most efficient at exploiting the available bandwidth of the copper wire transmission path. They achieve high bit-rate transmission by optimizing SNR performance through frequency-division multiplexing. Such systems will eventually allow delivery of sophisticated digital services, such as video on demand, multimedia, and internet access, over ordinary telephone wires without the added expense of laying additional wire or fiber optic cables.

Multicarrier QAM system encoding is shown in Figure 1. The N complex modulating vectors are given by

$$\mathbf{p}_n = \left[p_{n,0}\, p_{n,1} \cdots p_{n,N-1} \right] \tag{1}$$

where

$$p_{n,k} = \frac{1}{\sqrt{N}} e^{+j(2\pi/N)kn} \qquad k,n = 0,\cdots,N-1 \tag{2}$$

The complex input symbol, $\mathbf{D}_k=[D(0),...,(N-1)]^T$, must be Hermitian symmetric so that the transmitted signal, d_k, is real-valued. M-ary QAM is applied to each carrier according to the channel capacity within each subband. This requires channel probing to formulate the optimal bit allocation strategy. If the symbol period is defined as the period of the first-order harmonic frequency ($2\pi/N$), then the N parallel modulators can be considered equivalent to the inverse discrete Fourier transform (IDFT), with the advantage of efficient implementation via the IFFT.

With the absence of any channel characteristic and noise the QAM constellation point for each subband can be recovered with the DFT (FFT). However, all real channels convolve the transmitted signal with some channel response $h(n)$. We consider channel characteristics which can be modeled as an IIR filter with a finite number of poles and zeros. The significance of this assumption is that each N-sample encoded symbol can be prefixed with a cyclic extension to allow signal recovery using the cyclic convolution property of the discrete Fourier transform. If the length of the cyclic prefix is greater than or equal to the order of the numerator polynomial of the channel model, then by canceling the poles of the channel model with an adaptive equalizer, the constellation points in each subband can be recovered. The method requires frequency equalization following channel equalization and the DFT to compensate for complex scaling introduced by the filtering of the channel's numerator polynomial. This normalizes the DFT coefficients allowing uniform QAM decoding.

2. Adaptive equalization

Digital subscriber channels can usually be considered stationary since they are physically bundled underground. Therefore, we assume that an initial training period is sufficient to allow adaptive processing with a pseudo random bit sequence. Some tracking may be required, but we are concerned here with initial channel identification. The equation error adaptive filter structure is suitable for multicarrier channel equalization. Usually considered an alternative to output error IIR adaptive filters [3], the equation error filter provides a method of simultaneously identifying the channel's numerator polynomial, referred to as the target response, as well as its poles. The equation error adaptive filter is a two-input, single output FIR adaptive filter. As such, it retains most of the characteristics of FIR adaptive filters such as well behaved convergence properties and quadratic (unimodal) error surfaces. The equation error structure is shown in Figure 2. The equation output and equation error are given as

$$y_e(n) = \sum_{m=0}^{N-1} a(m)x(n-m) + \sum_{m=1}^{M-1} b(m)d(n-m) + \sum_{m=0}^{N-1} a(m)v(n-m) \tag{3}$$

$$e_e(n) = d(n) - y_e(n) \tag{4}$$

Equation (3) includes the additive noise source, and the equation output is not used during decoding. Its convenient to define the coefficient vector and the information vector as

$$\mathbf{w}(n) = [a(0) \cdots a(N-1)\ b(1) \cdots b(N-1)]^{\mathrm{T}} \tag{5}$$

$$\mathbf{u}(n) = [x(n) \cdots x(n-N+1)\ d(n-1) \cdots d(n-N+1)]^{\mathrm{T}} \tag{6}$$

so that equation (3) can be written as

$$y_e(n) = \mathbf{w}^{\mathrm{T}}(n)[\mathbf{u}(n) + \mathbf{v}(n)] \tag{7}$$

where

$$\mathbf{v}(n) = [v(n) \cdots v(n-N+1)\ 0 \cdots 0]^{\mathrm{T}} \tag{8}$$

With no channel noise ($\mathbf{v}(n)=0$) the mean-squared error is

$$\xi = \mathrm{E}\left[d^2(n)\right] + \mathbf{w}^{\mathrm{T}}(n)\mathrm{E}\left[\mathbf{u}(n)\mathbf{u}^{\mathrm{T}}(n)\right]\mathbf{w}(n) - 2\mathbf{w}^{\mathrm{T}}(n)\mathrm{E}[d(n)\mathbf{u}(n)] \tag{9}$$

from which the optimum coefficient vector follows through minimization as

$$\mathbf{w}_{\text{opt}} = \mathbf{R}_{uu}^{-1}\mathbf{p}_{du} \tag{10}$$

where

$$\mathbf{R}_{uu} = \mathrm{E}\left[\mathbf{u}(n)\mathbf{u}^{\mathrm{T}}(n)\right] \tag{11}$$

$$\mathbf{p}_{du} = \mathrm{E}[d(n)\mathbf{u}(n)] \tag{12}$$

With a matched-order equation error adaptive filter and no channel noise ($\mathbf{v}(n)=\mathbf{0}$), the minimum mean-squared equation error (MSEE) is zero, and the channel response can be identified from the optimum filter coefficients. The LMS algorithm applied to the equation error filter gives a simple method of descending the MSEE surface to reach the optimum coefficient solution in (10). The LMS algorithm for this filter structure is

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e_e(n)\mathbf{u}(n) \tag{13}$$

Two problems are readily identifiable with the method presented above. The first issue is that the additive channel noise was assumed to be zero. The expression for MSEE (9) must be modified to account for channel noise, assuming that the noise $v(n)$ and the channel output signal are uncorrelated, as follows

$$\xi = \mathrm{E}\left[d^2(n)\right] + \mathbf{w}^{\mathrm{T}}(n)\mathrm{E}\left[\mathbf{u}(n)\mathbf{u}^{\mathrm{T}}(n)\right]\mathbf{w}(n) - 2\mathbf{w}^{\mathrm{T}}(n)\mathrm{E}[d(n)\mathbf{u}(n)] + \mathbf{w}^{\mathrm{T}}(n)\mathrm{E}\left[\mathbf{v}(n)\mathbf{v}^{\mathrm{T}}(n)\right]\mathbf{w}(n) \tag{14}$$

Subsequent minimization of the MSEE in (14) results in the familiar biased coefficient solution for the equation error filter. Notice, however, that if $v(n)$ is additive white Gaussian noise (AWGN) then

$$\mathbf{R}_{vv} = \sigma_v^2\begin{bmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathbf{0}\end{bmatrix} \tag{15}$$

and the bias term in (14) becomes a constant if the coefficients in the channel path are constrained by normalization at each iteration. Therefore, the last term becomes σ_v^2, and the minimum MSEE coefficient solution is $\mathbf{w}_{\text{opt}}$.

The second problem is the slow rate of convergence of the LMS algorithm when applied to this particular problem. The severe attenuation characteristics of the channel introduce highly disparate eigenvalues in the autocorrelation matrix of the information signal. This can cause the filter to require millions of iterations to achieve satisfactory convergence, or the rate may be so slow that it seemingly won't converge at all. Orthogonalization can be applied to the sequential adaptive algorithm to increase the rate of convergence. The transform domain adaptive filter (TDAF) with channel power normalization [4] appears well-suited to decouple the adaptive modes and improve convergence performance for multicarrier gradient-based equalizers. This also allows a similar bias remedy for AWGN channels as discussed above.

The following (possibly complex) LMS algorithm can be applied to the TDAF

$$\mathbf{w}(n+1) = \mathbf{w}(n) + 2\mu e_e(n)\Lambda_u^{-2}\mathbf{c}^*(n) \tag{16}$$

where $\mathbf{c}(n)$ is the vector formed by premultiplying the concatenated input information vector $\mathbf{u}(n)$ by the unitary transform $\mathbf{T}$.

$$\mathbf{c}(n) = \mathbf{T}\mathbf{u}(n) \tag{17}$$

Channel normalization is realized by including the matrix $\Lambda_u^2=\text{diag}[\sigma_u^2(0)\sigma_u^2(1)...\sigma_u^2(N\text{-}1)]$, where each diagonal component is $\sigma_u^2(m)=E[|u(m)|^2]$.

Ideally, the Karhunen-Loeve Transform (KLT) is used to achieve optimal convergence, but this requires a priori knowledge of the input statistical properties. The KLT corresponding to the input autocorrelation matrix $\mathbf{R}_{uu}$ is constructed using as rows of $\mathbf{T}$ the orthonormal eigenvectors of $\mathbf{R}_{uu}$. Therefore, with unitary $\mathbf{Q}_u^H=[q_1 \ldots q_N]$ and $\Lambda_u=\text{diag}[\lambda_1 \ldots \lambda_N]$, the unitary similarity transformation is $\mathbf{R}_{uu}=\mathbf{Q}_u^{-1}\Lambda_u\mathbf{Q}_u$, and the KLT is given by $\mathbf{T}=\mathbf{Q}_u$. However, since the statistical properties of the input process are usually unknown and time varying, the KLT cannot be implemented in practice. Researchers have found that many fixed transforms do provide good orthogonalization for a wide class of input signals. Those include the Discrete Fourier Transform (DFT or FFT), the Discrete Cosine Transform (DCT), and the Walsh Hadamard Transform (WHT), all of which are suboptimal compared to the KLT. For example, the DFT provides only approximate channel decorrelation since it is well-known that a "sliding" DFT implements a parallel bank of overlapping band-pass filters with center frequencies evenly distributed over the interval $[0,2\pi]$. Furthermore, the DFT (or FFT) is hampered by the fact that it requires complex arithmetic. It is still a very effective method of orthogonalization . We propose using the DCT since it is real-valued and has a fast implementation. For the DCT, the matrix $\mathbf{T}$ is constructed as follows

$$t_{mk} = \begin{cases} \dfrac{1}{\sqrt{N}} & \begin{matrix} m=0 \\ k=0,\ldots,N\text{-}1 \end{matrix} \\ \sqrt{\dfrac{2}{N}}\cos\left(\dfrac{\pi(2k+1)m}{2N}\right) & \begin{matrix} m=1,\ldots,N\text{-}1 \\ k=0,\ldots,N\text{-}1 \end{matrix} \end{cases} \tag{18}$$

3. Simulation results

Simulation results illustrate the methods examined above. Figure 3 shows the magnitude response of a 18,000 foot pole-zero channel model used for all the simulations. Equalizer performance can be analyzed by examining both convergence plots and scatter diagrams. The training sequence consists of an encoded sequence with a complex symbol size of 128 with two bits per subband. Also, an artificial noise floor of -100 dB was used in each case. All of the adaptive filters were selected to be matched-order to allow sufficient degrees of freedom to compensate for the channel model.

The first set of simulations do not include channel noise. Figures 4 and 5 show scatter diagrams for the 1st and 40th subbands with no channel equalization. Intersymbol interference severely limits the ability to communicate effectively over the 63 subband DMT channel. The equation error equalizer with the LMS algorithm was introduced next. Figures 6, 7, and 8 show the learning curve and the scatter diagrams (after convergence) for the 1st and 40th subbands with no additive channel noise. The filter converges to the noise floor in 50,000 iterations, and the signal constellation is nearly perfect for all 63 subbands.

The previous experiment was then repeated with AWGN giving a 40 dB SNR. Figures 9, 10, and 11 show the convergence plot and the scatter diagrams for the 1st and 40th subbands. The channel noise causes several noticeable effects. First, it limits the convergence range by an amount equal to the SNR. Second, it biases the adaptive filter coefficients according to

equation (14). The scatter diagram for the first subband shows significant variance in the constellation points. Subband 40 appears unable to support traffic because of the degradation and noise enhancement.

Figures 12, 13, and 14 were produced using the equation error LMS adaptive filter with the white noise bias remedy. Some improvement is evident in the scatter diagram for the first subband. However, the higher subbands still appear unusable due to noise enhancement.

Finally, a convergence plot for the transform domain equation error filter (with DCT) is shown in Figure 15. The rate of convergence is clearly much faster than that of the LMS equation error filters.

4. Conclusions

Structures and algorithms for adaptive equalization of discrete multitone channels were presented. For HDSL/ADSL applications, proper equalization is essential to exploit the available bandwidth of an existing copper plant. It was shown that the equation error configuration provides a method of channel identification when combined with a suitable bias remedy. The equalizer was modified into a transform domain adaptive configuration to improve the rate of convergence, which can be prohibitively slow with real channels.

5. References

[1] J. Chow, J. C. Tu, and J. Cioffi, "A discrete multitone transceiver system for HDSL applications," *IEEE Journal on Selected Areas in Comm.* vol. 9, no. 6, August 1991.

[2] B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.

[3] J. Shynk, "Adaptive IIR filtering," *IEEE Signal Processing Magazine*, April, 1989.

[4] W. K. Jenkins, A. W. Hull, J. C. Strait, B. A. Schnaufer, and X. Li, *Advanced Concepts in Adaptive Signal Processing*. Kluwer Academic Press, 1996.

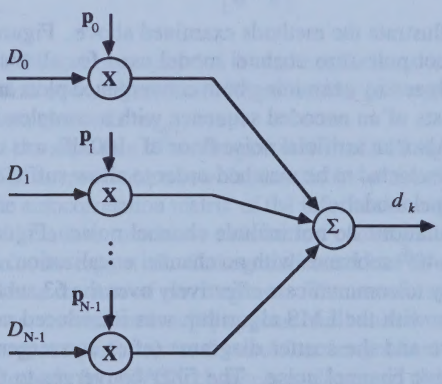


Figure 1. Multicarrier QAM.

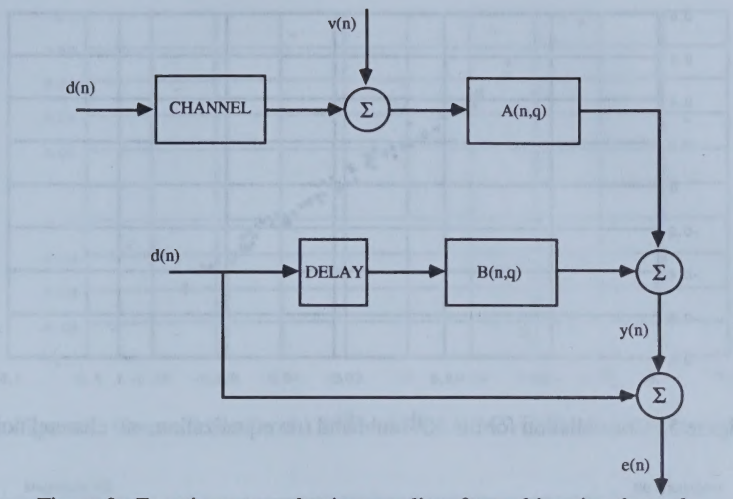


Figure 2. Equation error adaptive equalizer for multicarrier channels.

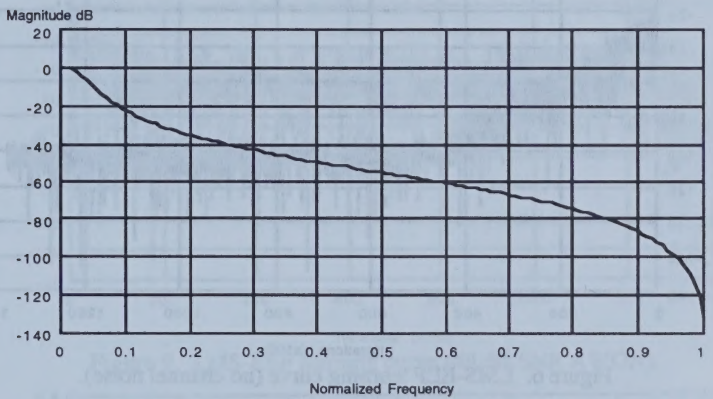


Figure 3. Magnitude response of the 18 kft channel pole-zero model.

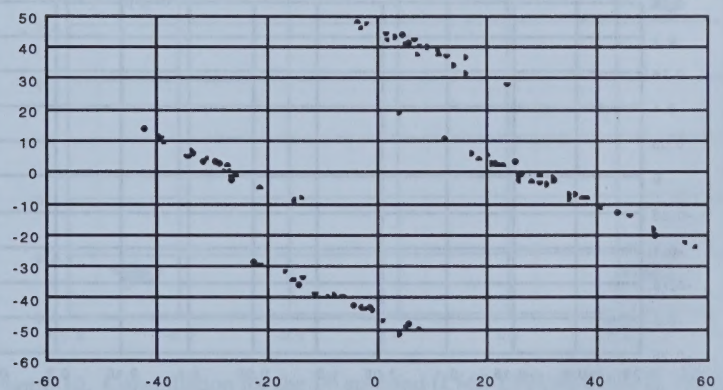


Figure 4. Constellation for the 1st subband (no equalization, no channel noise).

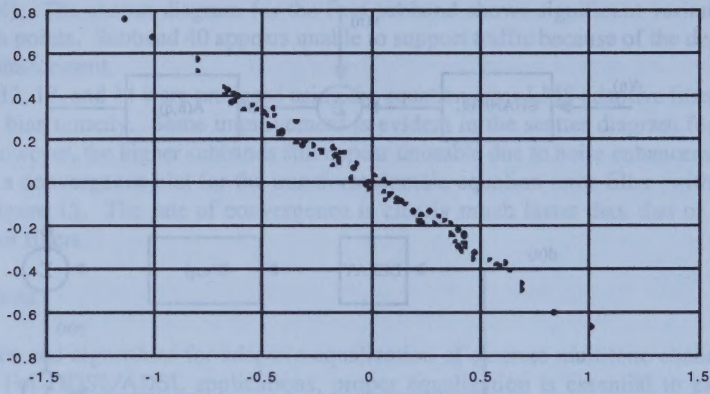


Figure 5. Constellation for the 40th subband (no equalization, no channel noise).

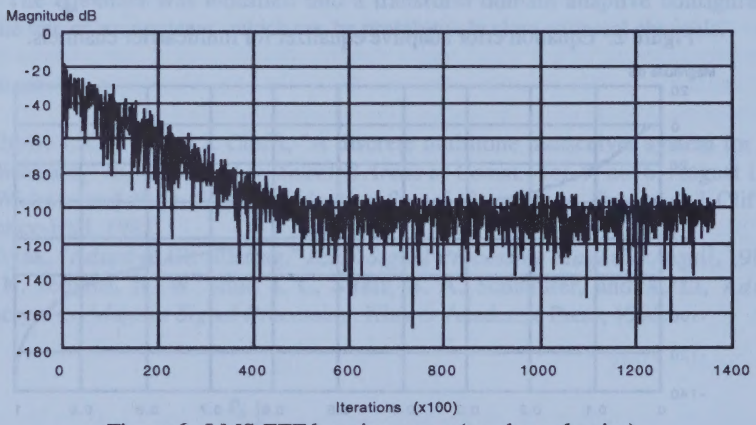


Figure 6. LMS-EEF learning curve (no channel noise).

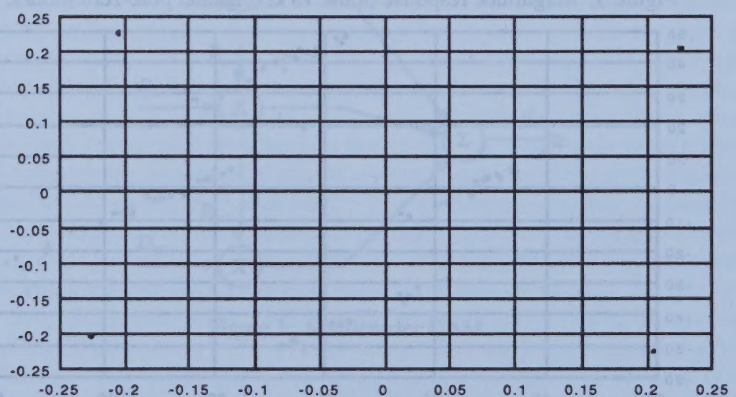


Figure 7. Constellation for the 1st subband (LMS-EEF, no channel noise).

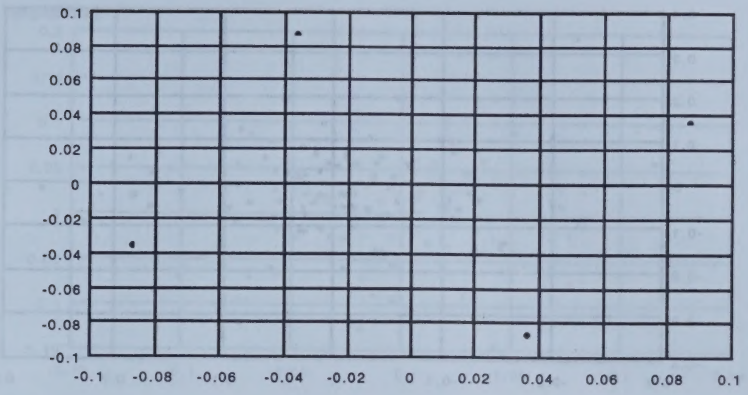


Figure 8. Constellation for the 40th subband (LMS-EEF, no channel noise).

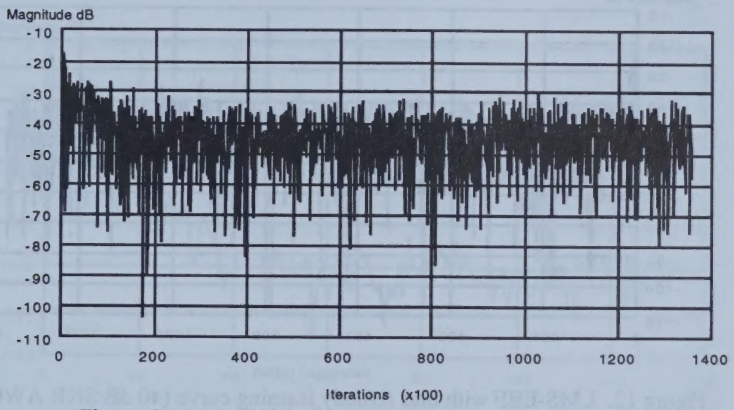


Figure 9. LMS-EEF learning curve (40 dB SNR AWGN).

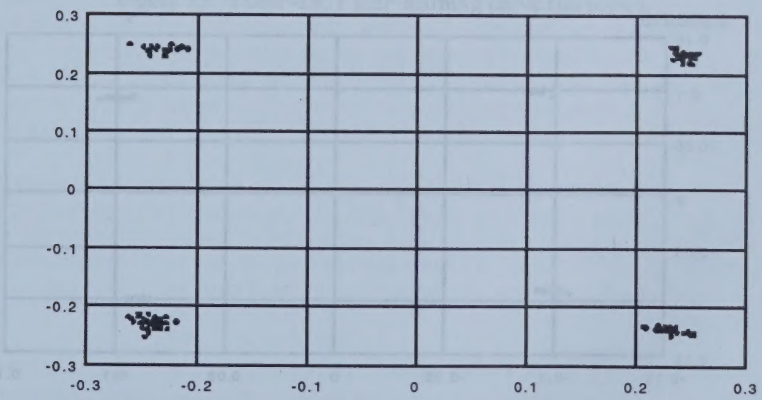


Figure 10. Constellation for the 1st subband (LMS-EEF, 40 dB SNR AWGN).

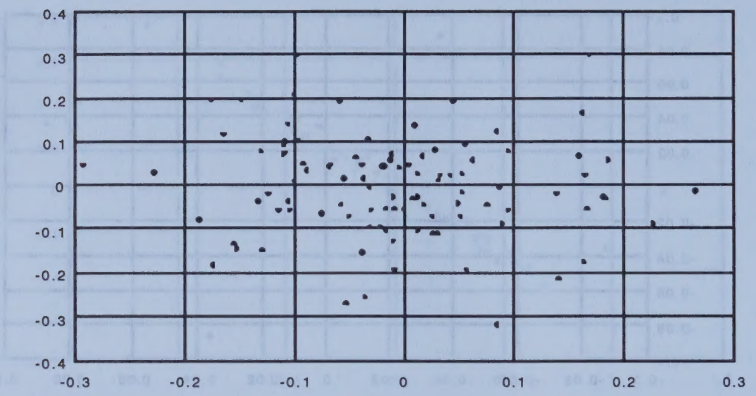


Figure 11. Constellation for the 40th subband (LMS-EEF, 40 dB SNR AWGN).

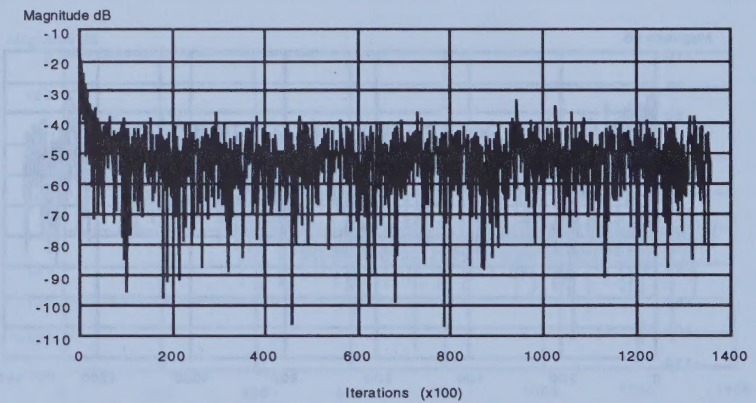


Figure 12. LMS-EEF with bias remedy learning curve (40 dB SNR AWGN).

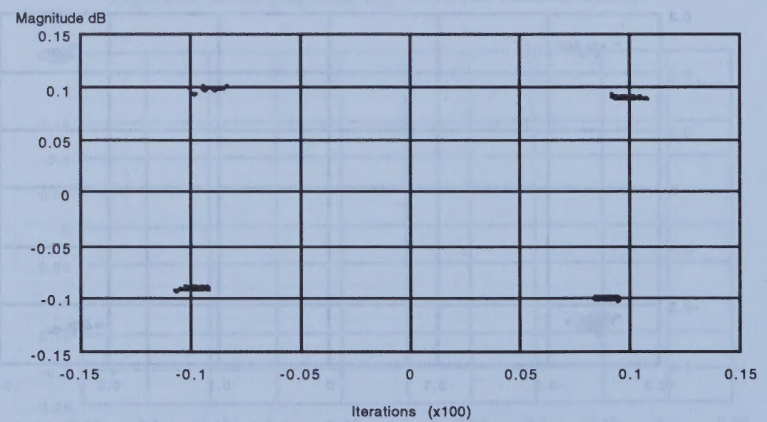


Figure 13. Constellation for the 1st subband (bias remedy, 40 dB SNR AWGN).

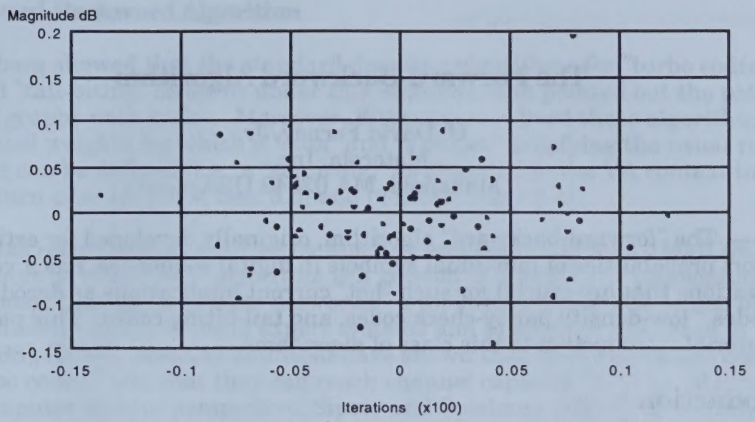


Figure 14. Constellation for the 40th subband (bias remedy, 40 dB SNR AWGN).

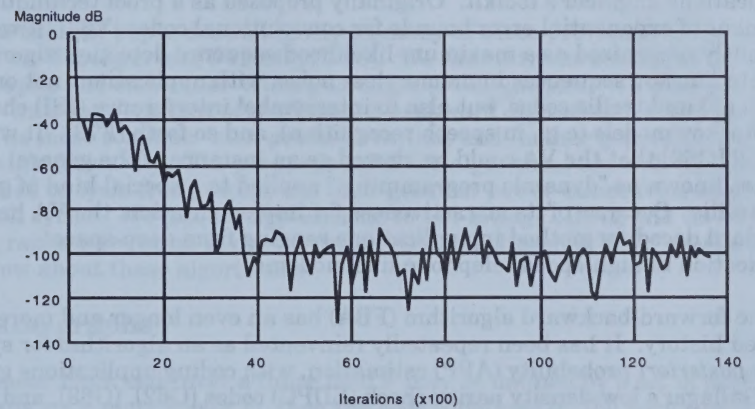


Figure 15. TDAF-DCT EEF learning curve (no noise).

The Forward-Backward Algorithm

G. David Forney, Jr.
Motorola, Inc.
Mansfield, MA 02048 USA

Abstract— The "forward-backward" algorithm, originally developed for estimation of *a posteriori* probabilities of individual symbols in digital sequences, has a variety of generalizations that are crucial for such "hot" current applications as decoding "turbo codes," low-density parity-check codes, and tail-biting codes. This paper gives a brief tutorial introduction to this class of algorithms.

1. Introduction

The Viterbi algorithm (VA) has become a standard tool in every communications engineer's toolkit. Originally proposed as a proof technique in the development of exponential error bounds for convolutional codes [V67], it was subsequently recognized as a maximum-likelihood sequence detection algorithm for finite-state Markov sequences in memoryless noise, with applications not only to convolutional and trellis codes, but also to intersymbol interference (ISI) channels, hidden Markov models (e.g., in speech recognition), and so forth [F73]. It was recognized [O69] that the VA could be viewed as an instance of the general class of techniques known as "dynamic programming," applied to a special kind of graph called a trellis. Because of its attractiveness for implementation, the VA has become the standard decoding method in applications ranging from deep-space communication to high-speed telephone-line modems.

The forward-backward algorithm (FBA) has an even longer and more convoluted history. It has been repeatedly reinvented as an algorithm for symbol-by-symbol *a posteriori* probability (APP) estimation, with coding applications going as far back as Gallager's low-density parity-check (LDPC) codes [G62], [G63], and for symbol-by-symbol APP estimation on ISI channels in a number of papers in the 1960's (e.g., [CH66], [AF70]). In the current coding literature it is most frequently called the BCJR algorithm, after [BCJR74]; however, in addition to the references already cited, [BCJR74] cites an earlier algorithm [BP66] from the statistical literature. The version of the FBA used in speech recognition is known as the "Baum-Welch" algorithm [R89]. There must be a large number of other references in various literatures.

The line of development upon which this paper is primarily based is as follows. In [T81], Tanner generalized Gallager's LDPC codes to codes on general bipartite graphs, here called "Tanner graphs," with two types of nodes representing symbols and parity checks, respectively. Tanner also developed the two algorithms that are here called "min-sum" and "sum-product," and proved that they converge on cycle-free graphs.

Wiberg *et al.* [WLK95], [W96] extended Tanner graphs to include invisible nodes, or "states," thus establishing a bridge to the trellis decoding literature. Here such graphs will be called "Tanner-Wiberg-Loeliger graphs." In this light the VA and the FBA can be seen as special cases of the "min-sum" and "sum-product" algorithms that arise when the underlying graph is a trellis. Other "soft-output" decoding algorithms such as the "soft-output Viterbi algorithm" [HH89] also fall in this class.

The Forward-Backward Algorithm

Wiberg showed that the standard decoding algorithms for "turbo codes," LDPC codes, and "tail-biting" codes fit under this umbrella, and pointed out the potential virtues of graphs with cycles. Moreover, Wiberg generalized these algorithms to any set of symbol weights for which a "sum" and "product" satisfying the usual rules of arithmetic can be defined (i.e., a "semiring," as observed in the VA context in [M96], which in turn cites [AHU74, Sec. 5.6] and [CLR90, Sec. 26.4]).

It has recently been realized by MacKay and others [MN96], [MMC96], [KF96] that the "sum-product" APP algorithm is a version of "probability propagation" or "belief propagation" in "Bayesian networks," a topic about which there has been much excitement in the fields of expert systems and artificial intelligence in the past decade [P88], [N90]. MacKay and Neal have shown that LDPC codes are competitive with "turbo codes," and that they can reach channel capacity [MN97]. Meanwhile, from a computer science perspective, Sipser and Spielman [SS96], [S96] have shown that LDPC codes offer a very attractive performance/complexity tradeoff, when used with the primitive "flipping" decoding algorithm also suggested by Gallager [G63].

This tutorial paper draws heavily on many of these previous papers, and in particular on Wiberg *et al.* [WLK95], [W96]. We first introduce codes defined on Tanner graphs and on Tanner-Wiberg-Loeliger (TWL) graphs. We then present in detail the "min-sum" variant of the forward-backward algorithm for "decoding" such graphs. The more common "sum-product" variant and further generalizations then follow straightforwardly. The properties of these algorithms are well understood when applied to cycle-free graphs, but their greatest power comes from applications to graphs with cycles. We discuss applications to tail-biting, LDPC, and turbo codes, all of whose graphs have large-diameter cycles. Our conclusion is that "everyone" should know about these algorithms.

2. Codes on graphs

A linear block code over a finite field F may be defined by a set of parity checks. Let H be any r x n matrix over F; then the code C defined by the parity-check matrix H is the set of all n-tuples $\mathbf{x}$ over F such that $H\mathbf{x} = \mathbf{0}$ (with $\mathbf{x}$ as a column vector).

For example, the 3 x 7 parity-check matrix

$$H = \begin{matrix} 1110000 \\ 1001100 \\ 1000011 \end{matrix}$$

defines a simple binary block code C, namely, the set of $2^4 = 16$ binary length-7 codewords $\mathbf{x} = (x_1, x_2, \ldots, x_7)$ that satisfy the three parity checks

$$\begin{aligned} x_1 + x_2 + x_3 &= 0; \\ x_1 + x_4 + x_5 &= 0; \\ x_1 + x_6 + x_7 &= 0, \end{aligned}$$

where addition is "mod-2" addition in the binary field F_2. This code is denoted as a (7, 4, 2) code since it has length 7, dimension 4 (as a subspace of $(F_2)^7$), and minimum Hamming distance 2 (e.g., 0000000 and 0110000 are codewords).

The Forward-Backward Algorithm

A Tanner graph [T81] is a graphical representation of a code corresponding to a set of parity checks that specify the code. Each symbol is represented by a symbol node, and each parity check is represented by a parity-check node. Each parity-check node is connected by an edge to the symbols that it checks. A Tanner graph is thus a bipartite graph in which symbol nodes are connected only to parity-check nodes and *vice versa*.

For example, Figure 1 illustrates the Tanner graph corresponding to the (7, 4, 2) code above. Symbol nodes are represented by black circles, and parity-check nodes are represented by circles that mimic the notation "⊕" for mod-2 addition.

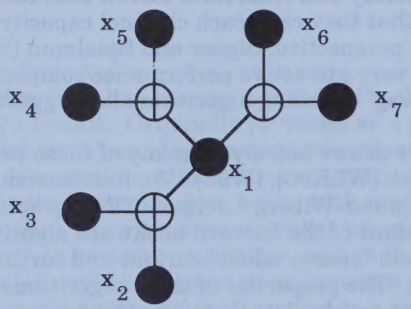


Figure 1. Tanner graph for (7, 4, 2) binary code.

A Tanner graph defines a code in terms of local constraints, which suggests a kind of neural network. The constraints (parity checks) are very simple: certain configurations of values of the attached symbol nodes are allowed, and the rest are not. In the terminology of behavioral system theory [WLK95], the code is defined by its local behaviors. More generally, as observed in [W96], [FK96], a Tanner graph defines a special kind of Markov random field in which the local probability distributions are equiprobable over a subset of the possible configurations.

From a probabilistic viewpoint, a Tanner graph defines dependency chains in a "Bayesian network" [P88]. In particular, it defines instances of conditional independence. For example, in Figure 1, given x_1, the pairs of variables (x_2, x_3), (x_4, x_5) and (x_6, x_7) become conditionally independent of each other. In this respect the variable x_1 acts like a "state" in the sense of system theory, or a "sufficient statistic" in the sense of statistical inference.

A key contribution of Wiberg *et al.* [W96], [WLK95] was the introduction of a third type of node, representing an invisible "state." This innovation greatly expands the kinds of graphs and dependency chains by which a code can be represented, and connects Tanner graphs with trellis representations. We call such a graph a "Tanner-Wiberg-Loeliger graph."

For example, Figure 2(a) shows a minimal trellis code for the (7, 4, 2) block code example, with "state spaces" of sizes {1, 2, 4, 2, 4, 2, 2, 1}. Figure 2(b) shows the corresponding TWL graph. Here the graph is linear, representing a "time axis," and the states are represented by open circles of different sizes. The "parity checks"

represent the constraints governing a particular trellis section; only certain pairs of neighboring states ("transitions") are allowed, and each allowed pair uniquely determines an associated code symbol.

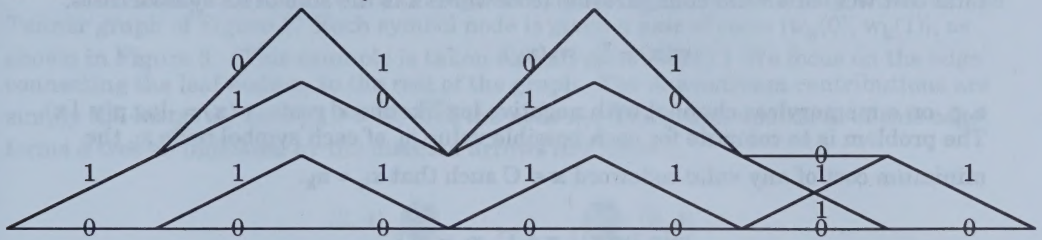


Figure 2(a). Minimal trellis diagram for (7, 4, 2) code.

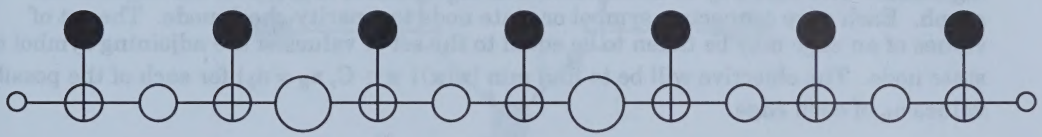


Figure 2(b). Tanner-Wiberg-Loeliger graph for (7, 4, 2) code.

There has been an enormous amount of work recently on trellis representations of linear block codes, which we will not attempt to summarize here (*cf.* the November 1996 *IEEE Transactions of Information Theory* Special Issue on Codes and Complexity). The central facts are that, given a coordinate ordering, every linear code has an essentially unique minimal trellis representation; the sizes of the state spaces depend significantly on the ordering; but even with the most efficient orderings, trellis complexity increases exponentially with block length.

The Tanner-Wiberg-Loeliger graph corresponding to a trellis diagram is obviously cycle-free. It follows that every linear block code can be represented by a cycle-free TWL graph. However, this may not be the simplest cycle-free representation; for example, the Tanner graph of Figure 1 is for most purposes a simpler cycle-free graph representation of the (7, 4, 2) code than any trellis representation as in Figure 2. Furthermore, as we shall see later, graph representations with cycles may be even simpler.

3. Decoding cycle-free graphs

The forward-backward algorithm is a straightforward solution to certain decoding problems for finite cycle-free Tanner-Wiberg-Loeliger graphs with finite-valued symbol and state nodes, including graphs representing trellises.

3.1 The min-sum algorithm

We first consider the following problem. Each possible value a_k of each symbol node x_k is assigned a cost $w_k(a_k)$; for example, a negative log likelihood

$$w_k(a_k) = -\log p(y_k \mid x_k = a_k),$$

where y_k is the received symbol corresponding to the code symbol x_k. (The state nodes could also be assigned costs, but for now we will assume that states are cost-free.) The total cost $w(\mathbf{x})$ of a valid configuration (codeword) $\mathbf{x}$ is the sum of its symbol costs:

$$w(\mathbf{x}) = \sum_k w_k(x_k);$$

e.g., on a memoryless channel with negative log likelihood costs, $w(\mathbf{x}) = -\log p(\mathbf{y} \mid \mathbf{x})$. The problem is to compute for each possible value a_k of each symbol node x_k the minimum cost of any valid codeword $\mathbf{x} \in C$ such that $x_k = a_k$:

$$\min \{w(\mathbf{x}) \mid \mathbf{x} \in C, x_k = a_k\}.$$

The solution to this problem, for finite cycle-free graphs, is the "min-sum" algorithm. The best way to think about this algorithm is to focus on the edges of the graph. Each edge connects a symbol or state node to a parity-check node. The set of values of an edge may be taken to be equal to the set of values of the adjoining symbol or state node. The objective will be to find $\min \{w(\mathbf{x}) \mid \mathbf{x} \in C, x_k = a_k\}$ for each of the possible values a_k of each edge.

The key observation is that a cut through any edge of a cycle-free graph cuts the graph into two disjoint parts, say "upstream" and "downstream." The cost $w(\mathbf{x})$ of any codeword $\mathbf{x}$ may correspondingly be expressed as the sum

$$w(\mathbf{x}) = w(\mathbf{x}^+) + w(\mathbf{x}^-)$$

of the upstream and downstream contributions $w(\mathbf{x}^+)$ and $w(\mathbf{x}^-)$, respectively. The min-sum algorithm computes these two contributions separately.

In a finite cycle-free graph, the part of the graph upstream or downstream from any edge forms a finite tree with the edge at the root. The upstream or downstream contributions may be computed by starting from the leaves of this tree and working downwards toward the root. If the tree is finite, then the number of computations required to reach the root must be finite.

The update rules for these computations are as follows. For an edge downstream of a symbol or state node, for each possible a_k, the upstream contribution is the sum of the upstream contributions coming into the symbol or state node plus the local contribution (cost) of the node itself:

$$\min \{w(\mathbf{x}^+) \mid \mathbf{x}^+ \in C^+, x_k = a_k\} = w_k(a_k) + \sum_j \min \{w(\mathbf{x}_j^+) \mid \mathbf{x}_j^+ \in C_j^+, x_k = a_k\},$$

where C^+ is the set of valid upstream configurations given $x_k = a_k$, j indexes the edges upstream of the node, $\mathbf{x}_j^+$ is the part of the graph upstream of edge j, and C_j^+ is the set of valid configurations upstream of edge j given that $x_k = a_k$.

For an edge downstream of a parity-check node, for each possible a_k, the upstream contribution is the minimum of the sum of all upstream contributions over configurations consistent with a_k:

$$\min \Sigma_j \{w(\mathbf{x}_j^+) \mid \mathbf{x}_j^+ \in C_j^+, x_k = a_k\}.$$

For example, we shall work through an example of the min-sum algorithm for the Tanner graph of Figure 1. Each symbol node is given a pair of costs $\{w_k(0), w_k(1)\}$, as shown in Figure 3. (This example is taken directly from [W96].) We focus on the edge connecting the leaf node x_7 to the rest of the graph. The downstream contributions are simply the costs $(w_7(0), w_7(1)) = (2, 6)$ of the node x_7. The graph upstream of this edge forms a tree as indicated by the directed arrows in Figure 3.

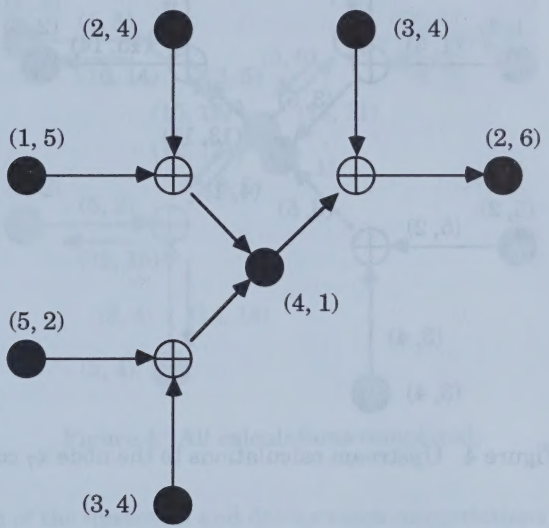


Figure 3. Tanner graph of Fig. 1 with node costs and tree upstream of x_7.

The upstream contributions from the edges connected to the leaf nodes $x_2, \ldots, x_6$ are simply the associated symbol node costs. Next, to compute the upstream contribution from the edge connecting x_1 to the parity-check node connected to x_2 and x_3, for $x_1 = 0$, we must consider the two possibilities $(x_2, x_3) = (0, 0)$ or $(1, 1)$, which have costs $3 + 5 = 8$ and $4 + 2 = 6$, respectively; the minimum is $(1, 1)$ with cost 6. Similarly, for $x_1 = 1$ the minimum of $3 + 2$ and $4 + 5$ is 5. So the costs for this edge are $(6, 5)$. Similarly, the costs for the the edge connecting x_1 to the parity-check node connected to x_4 and x_5 are

$$(\min \{1 + 2, 5 + 4\}, \min \{1 + 4, 5 + 2\}) = (3, 5).$$

Now we have all the information that we need to compute the upstream contributions to the edge downstream of x_1, which are

$$(4, 1) + (6, 5) + (3, 5) = (13, 11).$$

From this we can find the upstream contributions out of the last parity-check node,

$$(\min\{13 + 3, 11 + 4\}, \min\{13 + 4, 11 + 3\}) = (15, 14).$$

Summing these with the downstream contributions (2, 6) to this edge, we obtain as our final result the minimum costs (17, 20) of codewords with $x_7 = 0$ and 1, respectively. These calculations are summarized in Figure 4.

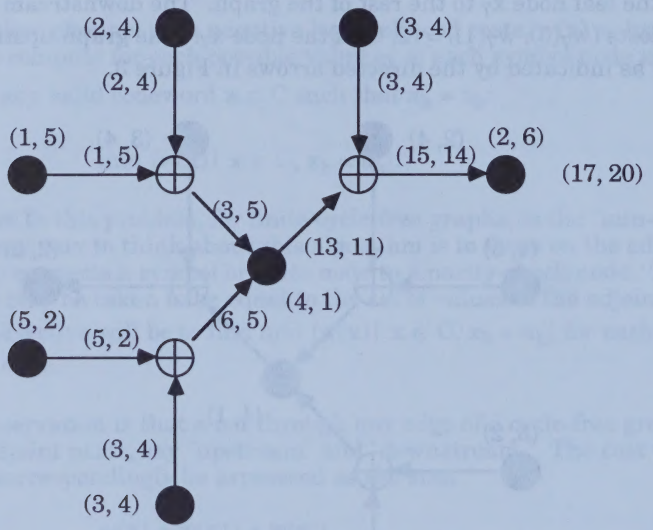


Figure 4. Upstream calculations to the node x_7 completed.

We pause to make a few remarks at this point. First, if our real interest is the minimum-cost codeword, then we may simply take the minimum of $\min\{w(\mathbf{x}) \mid \mathbf{x} \in C, x_k = a_k\}$ over all a_k, for any k. In this case, from the costs at x_7, the minimum-cost codeword has cost equal to $\min(17, 20) = 17$, and thus has $x_7 = 0$. The entire minimum-cost codeword may now be determined if the "decisions" made at each minimization step have been stored, via a "trace-back" procedure through the graph. For a linear graph corresponding to a trellis, this is nothing more nor less than the standard Viterbi algorithm. For a more general cycle-free graph, this is simply a dynamic programming solution to the minimum-cost problem.

The "min-sum" algorithm tells us more, however; it computes the relative costs of all possible symbol values, which may be used as a kind of "soft decision." Applied to a trellis, this leads to a "soft-decision Viterbi algorithm" like that of [HH89].

If all nodes are binary-valued, then it suffices to know a single number, the difference between the costs of 0 and 1. This leads to certain simplifications in the update rules whose origin goes back at least as far as [G63]. In particular, the least-cost output from a parity-check node is the sum of the least-cost inputs, its value is the mod-2 sum of the least-cost input values, and the cost difference coming out is the minimum of the cost differences coming in. Many instances of these simplified rules may be seen in Figure 4.

The Forward-Backward Algorithm

Now let us return to the computation of the all pairs of costs for all edges in the Figure 3 graph. One way of completing this computation is to go back and compute all costs in the downstream direction working back from the root; hence the name "forward-backward algorithm." The completed computation is shown in Figure 5.

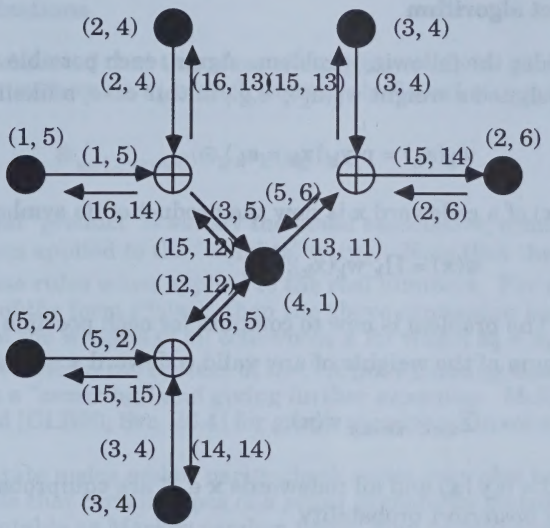


Figure 4. All calculations completed.

Taking the sum of the upstream and downstream computations at each edge, we find that one of the values of each pair is 17, the cost of the minimum-cost codeword, namely 1010110. In addition, we find that the magnitudes of the cost differences are (1, 1, 3, 2, 1, 1, 3), which tells us that in the least-cost codeword the "decisions" on x_3 and x_7 are most reliable, while those on x_1, x_2, x_5 and x_6 are least reliable.

An interesting sidelight is that since the two downstream contributions out of x_1 towards (x_2, x_3) are equal, the rest of the graph provides no information for the decisions about (x_2, x_3). These two symbols are therefore decided solely according to their local costs as (0, 1), which in turn implies $x_1 = 1$. The pairs (x_4, x_5) and (x_6, x_7) are then decided simply as the minimum-cost pair of the two odd-parity pairs. These observations illustrate the kinds of simplifications that can arise from conditional independence.

We now consider how best to order the computations. It is clear that each upstream and downstream contribution to each edge needs to be computed only once, and that this computation can be performed when and only when all of the relevant "input" contributions are available. The minimum total number of computations required is therefore equal to twice the total number of edge values.

The computations are all local. We can therefore think of them as being implemented in "neural network" style with two processors per edge (upstream and downstream). Computations can simply be allowed to propagate from the leaf nodes, in

asynchronous or synchronous fashion. In a synchronous computation, the maximum number of clock cycles required is the maximum depth of any tree upstream or downstream of any edge; e.g., 4 cycles in our continuing example. For a finite graph, convergence of the algorithm in a finite time is therefore guaranteed.

3.2 The sum-product algorithm

Next we consider the following problem. Again, each possible value a_k of each symbol node x_k is assigned a weight $w_k(a_k)$; e.g., in this case, a likelihood

$$w_k(a_k) = p(y_k \mid x_k = a_k).$$

The total weight $w(\mathbf{x})$ of a codeword $\mathbf{x}$ is now the product of its symbol weights:

$$w(\mathbf{x}) = \prod_k w_k(x_k);$$

e.g., $w(\mathbf{x}) = p(\mathbf{y} \mid \mathbf{x})$. The problem is now to compute for each possible value a_k of each symbol node x_k the sum of the weights of any valid codeword $\mathbf{x} \in C$ such that $x_k = a_k$:

$$\sum_{\mathbf{x} \in C,\, x_k = a_k} w(\mathbf{x}).$$

Note that when $w(\mathbf{x}) = p(\mathbf{y} \mid \mathbf{x})$ and all codewords $\mathbf{x} \in C$ are equiprobable, this sum is proportional to the *a posteriori* probability

$$p(x_k = a_k \mid \mathbf{y}) \sim \sum_{\mathbf{x} \in C,\, x_k = a_k} p(\mathbf{y} \mid \mathbf{x}).$$

The solution to this problem, for finite cycle-free graphs, is the "sum-product" algorithm. The "sum-product" algorithm is merely the "min-sum" algorithm with "min" replaced by "sum" and "sum" replaced by "product." In particular, the rules for the update computations are as follows. For an edge downstream of a symbol or state node, for each possible a_k, the upstream contribution is the product of the upstream contributions coming into the symbol or state node times the local contribution (weight) of the node itself:

$$\sum_{\mathbf{x}^+ \in C^+,\, x_k = a_k} w(\mathbf{x}^+) = w_k(a_k) \left(\prod_j \sum_{\mathbf{x}_j^+ \in C_j^+,\, x_k = a_k} w(\mathbf{x}_j^+)\right).$$

For an edge downstream of a parity-check node, for each possible a_k, the upstream contribution is the sum over all products of upstream contributions from configurations consistent with a_k:

$$\sum_{\mathbf{x}_j^+ \in C_j^+,\, x_k = a_k} \left(\prod_j w(\mathbf{x}_j^+)\right).$$

The properties of the sum-product algorithm are precisely the same as the properties of the min-sum algorithm. In particular, it is guaranteed to converge in the same number of steps on a finite cycle-free graph, and the number of computations is the same. Nonetheless it has traditionally been considered less attractive than the min-sum algorithm, partly because of the requirement for multiplications, but probably more significantly because there is no "forward-only" version analogous to the Viterbi algorithm or its generalizations for the "min-sum" algorithm. Thus the "min-sum"

algorithm has often been proposed as a simplification or approximation of the "sum-product" algorithm. As one notable example, in his thesis Ungerboeck [U71] proposed the "min-sum" algorithm as an approximation to the known symbol-by-symbol APP decoding algorithms for ISI channels (e.g., [CH66], [AF70]).

3.3 Further generalizations

Wiberg [W96] observed that this class of algorithms can be generalized to solve the problem of finding for each value a_k of each node x_k the "sum" of "products"

$$\oplus_{\mathbf{x} \in C,\, x_k = a_k} (\otimes_{k'} w_{k'}(x_{k'})),$$

where the "sum" $\oplus$ and "product" $\otimes$ satisfy the usual associative, commutative and distributive rules when applied to the "weights" $w_k(x_k)$. Note that the operations "min" and "sum" satisfy these rules when applied to the real numbers. For another example, the weights could be of the form $z^{w_k(a_k)}$; then the above expression becomes the generating function of the weights of all codewords $\mathbf{x}$ for which $x_k = a_k$. McEliece has made a similar observation in the context of the VA [M96], noting that the resulting algebraic structure is a "semiring," and giving further examples. McEliece further cites [AHU74, Sec. 5.6] and [CLR90, Sec. 26.4] for graph algorithms involving semirings.

Of course the state nodes and/or parity-check nodes may also be given weights. Wiberg [W96] suggests that certain types of *a priori* probability distributions $p(\mathbf{x})$ (namely, distributions representable as Markov random fields) may be represented in this way.

4. Decoding graphs with cycles

Since the update equations of these algorithms are local, it is clear that they could be applied in some fashion to graphs that are not cycle-free.

Why would one want to consider such graphs? We shall show, specifically for tail-biting codes, that graph representations with cycles can be significantly simpler than cycle-free graph representations, and thus can yield simpler decoding algorithms. Moreover, the most powerful codes known, namely turbo codes and low-density parity-check codes, are naturally represented by graphs with cycles, and are naturally decoded by versions of these algorithms.

When cycles are present, it is not immediately clear how to generalize these algorithms. Issues that must be addressed include:

- Initialization
- Order of updating
- Convergence
- Stopping rule

Theoretically, very little is known about these issues. McEliece et al. [MRC95] have given several examples in which these iterative algorithms fail on graphs with cycles. Empirically, however, there is often a "natural" approach to these issues, and the resulting algorithms often work very well, provided that the cycles are long enough that cyclical dependencies die out.

4.1 Tail-biting codes

It has long been known that there are efficient "tail-biting" (or "quasi-cyclic," or "double circulant") representations of certain well-known block codes. A tail-biting trellis can have many fewer states than a standard trellis.

For example, a conventional trellis for the (24, 12, 8) binary Golay code must have at least 256 states at its midpoint [M88]. The proof is essentially that only 2 of the 12 generators of the code can be supported by the first half of the coordinates, since the largest value of k for which there exists a (12, k, 8) code is k = 2. Similarly, at most 2 of the generators can be supported by the second 12 coordinates. Consequently at least 8 generators must span the midpoint, which by the theory of minimal trellises implies that there must be at least $2^8 = 256$ states at the midpoint.

In contrast, a tail-biting trellis could have as few as 16 states [W96]. Indeed, we conjecture that there exists a generator matrix for the Golay code of the following form:

xxxx	xxxx	xxxx	0000	0000	0000
xxxx	xxxx	xxxx	0000	0000	0000
0000	xxxx	xxxx	xxxx	0000	0000
0000	xxxx	xxxx	xxxx	0000	0000
0000	0000	xxxx	xxxx	xxxx	0000
0000	0000	xxxx	xxxx	xxxx	0000
0000	0000	0000	xxxx	xxxx	xxxx
0000	0000	0000	xxxx	xxxx	xxxx
xxxx	0000	0000	0000	xxxx	xxxx
xxxx	0000	0000	0000	xxxx	xxxx
xxxx	xxxx	0000	0000	0000	xxxx
xxxx	xxxx	0000	0000	0000	xxxx

Such a generator matrix could be implemented by a rate-2/4 convolutional encoder, with 12 input bits entered cyclically so that the final encoder state equals the initial state. This would imply a tail-biting TWL graph representation of the form shown in Figure 5, where each symbol node represents 4 bits and each state has 16 possible values.

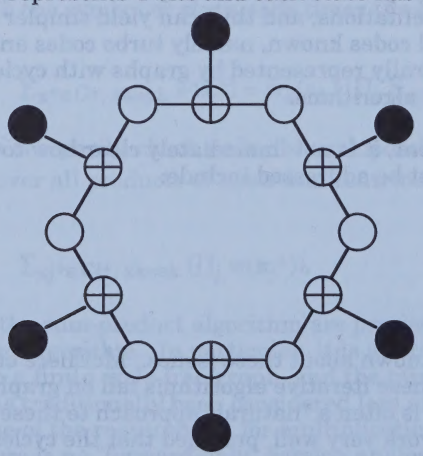


Figure 5. Tanner-Wiberg-Loeliger graph for tail-biting trellis.

The Forward-Backward Algorithm

The standard method of decoding a tail-biting trellis code is to initialize all values of a starting state to zero, and then to execute the Viterbi algorithm for a few circuits around the loop, at which point a clear "winning" path should have emerged as an "attractor." Empirically, such an approach works almost as well as maximum likelihood sequence decoding, and it is considerably simpler than a single VA pass through a much larger (e.g., 256-state) trellis.

The algorithms discussed here and the symmetry of tail-biting trellises suggest other approaches that might work even better; e.g., initializing all values of all states to zero, and updating all states in both directions simultaneously.

4.2 Low-density parity-check codes

Low-density parity-check (LDPC) codes, now seen as the grandfather of all these graph codes and decoding algorithms, were introduced by Gallager in his thesis [G63]. In a (j, k) LDPC code, every code symbol is checked by precisely j parity checks, and every parity check checks precisely k code symbols. Gallager introduced both a simple "flipping algorithm" (flip a bit if the majority of its checks so indicate), as well as an algorithm equivalent to the sum-product APP algorithm, with all symbols updated simultaneously. The algorithm stops when all *a posteriori* probabilities approach 0 or 1.

Although limited by the computing technology of that day, Gallager was able to demonstrate reliable decoding at rates above the computational cut-off rate R_0 of noisy binary symmetric channels (BSCs).

Sipser and Spielman [SS96] and Spielman [S96] have recently constructed very large LDPC codes using expander graphs, and shown that with the simple "flipping algorithm" reliable performance can be achieved at rates near R_0 on noisy BSCs, with decoding complexity only linear in block length.

Even more recently, MacKay and Neal [MN96] have shown that LDPC codes can achieve almost the performance of turbo codes on additive white Gaussian noise (AWGN) channels, using a number of iterations of the sum-product algorithm. Furthermore, they have shown that in principle LDPC codes can achieve channel capacity [MN97].

4.3 Turbo codes

Turbo codes are the highest-performance codes known for memoryless channels such as the AWGN channel.

While by now there are many variants, the standard example remains that of the original paper of Berrou *et al.* [BGT93], shown in Figure 6. An information sequence is fed into a systematic recursive rate-1/2 convolutional encoder, which generates a first check sequence. The same information sequence is interleaved in a long, pseudo-random interleaver, and then fed into a second identical encoder that generates a second check sequence. The information sequence and both check sequences are transmitted over the channel, for an overall code rate of 1/3.

The decoding procedure is as follows. The original channel likelihoods are used in a first decoding of the first code, which produces updated reliability information ("soft decisions") about all information symbols. Initially the full forward-backward sum-product algorithm was used for this decoding, but it has been found that

simplifications such as a soft-output Viterbi algorithm also can work well. This updated reliability information is then passed to a second decoder, which decodes the second code and produces updated reliabilities for the information symbols for subsequent use by the first decoder. The decoders alternate in this way for 10-20 iterations or until convergenc is achieved. Performance within 1 dB of the Shannon limit is now routinely demonstrated.

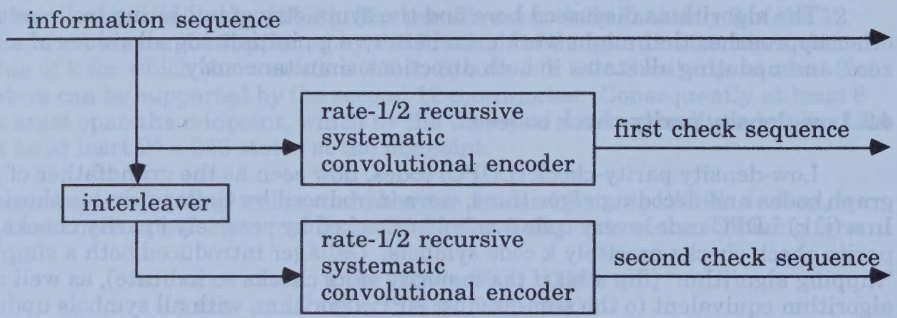


Figure 6. Standard turbo code encoder.

The Tanner-Wiberg-Loeliger graph of such a code, shown in Figure 7, is one of the great contributions of the Wiberg thesis [W96], since it gives much insight into turbo cod structure. Each of the component codes is cycle-free, but the interleaver introduces interconnections than can and do lead to cycles. However, if the interleaver is long, the cycles are very long. As Wiberg says [W96], "the amazing performance of turbo codes is primarily due to the cycle structure of the [TWL] graph..."

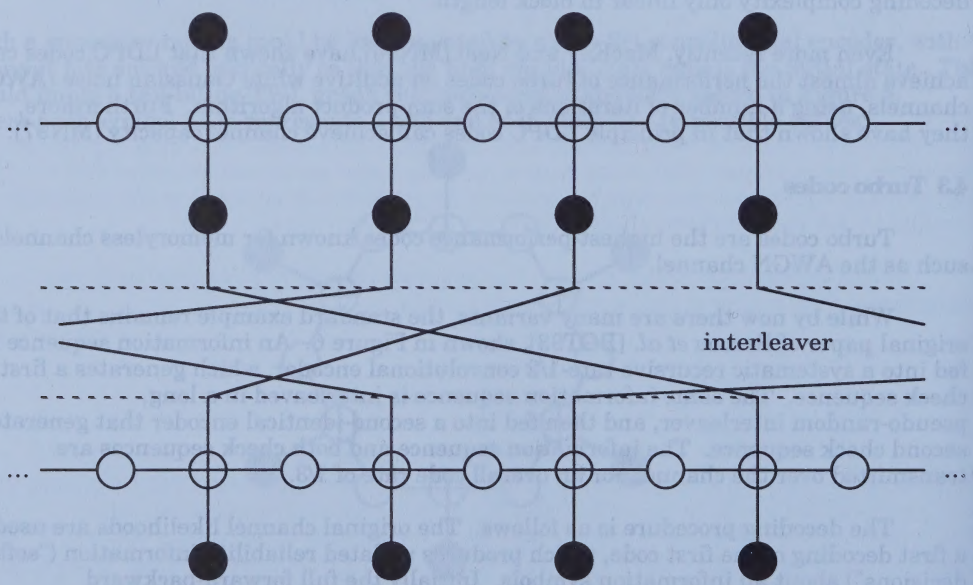


Figure 7. Tanner-Wiberg-Loeliger graph of a turbo code.

5. Conclusion

The forward-backward algorithm in both its min-sum and sum-product form should be part of the toolkit of every communications engineer. It is not much more complicated either conceptually or in practice than the Viterbi algorithm, at least for cycle-free graphs. Applied to graphs with cycles, it is the basis of the decoding methods that make feasible the best-performing codes currently known.

It is also used (and has been for more than 30 years) in cognate fields ranging from symbol-by-symbol APP detection of digital sequences in the presence of ISI, to tracking of hidden Markov models in speech recognition, to "belief propagation" in expert systems. In other words, it has become a standard tool of statistical inference. Given all the current activity in this field, further applications are sure to arise.

Very little is known yet in general about the analysis and synthesis of codes based on graphs. We are in the early days when a few strikingly good codes based on *ad hoc* constructions are known, but where no general theory of graphical representations of known codes or of graphical constructions of new classes of codes exists. Coding theorists will surely not be slow to pick up this gauntlet.

Acknowledgments

I am grateful to Brendan Frey, Bob Gallager, Ralf Kötter, Frank Kschischang, David MacKay, Bob McEliece, Mitch Trott, Gottfried Ungerboeck, and above all Andi Loeliger and Niclas Wiberg for sharing with me their knowledge of this subject, its history and its applications.

References

[AF70] K. Abend and B.D. Fritchman, "Statistical detection for communication channels with intersymbol interference," *Proc. IEEE*, vol. 58, pp. 779-785, May 1970.

[AHU74] A.V. Aho, J.E. Hopcroft and J.D. Ullman, *The Design and Analysis of Computer Algorithms*. Reading, MA: Addison-Wesley, 1974.

[BCJR74] L. Bahl, J. Cocke, F. Jelinek and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 284-287, Mar. 1974.

[BP66] L.E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite-state Markov chains," *Ann. Math. Stat.*, vol. 37, pp. 1559-1563, 1966.

[BGT93] C. Berrou, A. Glavieux and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo codes (1)," *Proc. ICC '93*, Geneva, pp. 1064-1070, June 1993.

[CH66] R.W. Chang and J.C. Hancock, "On receiver structures for channels having memory," *IEEE Trans. Inform. Theory*, vol. IT-12, pp. 463-468, Oct. 1966.

[CLR90] T.H. Cormen, C.E. Leiserson and R.L. Rivest, *Introduction to Algorithms*. Cambridge, MA: MIT Press, 1990.

[F73] G.D. Forney, Jr., "The Viterbi algorithm," *Proc. IEEE*, vol. 61, pp. 268-278, Mar. 1973.

[FK96] B.J. Frey and F.R. Kschischang, "Early detection and trellis splicing: Reduced-complexity soft iterative decoding," preprint, 1996.

[G62] R.G. Gallager, "Low-density parity-check codes," *IRE Trans. Inform. Theory*, vol. IT-8, pp. 21-28, Jan. 1962.

[G63] R.G. Gallager, *Low-Density Parity-Check Codes*. Cambridge, MA: MIT Press, 1963.

[HH89] J. Hagenauer and P. Hoeher, "A Viterbi algorithm with soft-decision outputs and its applications," *Proc. GLOBECOM '89*, Dallas, TX, pp. 47.1.1-7, Nov. 1989.

[MMC96] D.J.C. MacKay, R.J. McEliece and J.-F. Cheng, "Turbo decoding as an instance of Pearl's 'belief propagation' algorithm," submitted to *IEEE J. Select. Areas Commun.*, Sept. 1996.

[MN96] D.J.C. MacKay and R.M. Neal, "Near Shannon limit performance of low-density parity-check codes," *Elect. Lett.*, 1996, to appear.

[MN97] D.J.C. MacKay and R. M. Neal, "Good error-correcting codes based on very sparse matrices," preprint, 1996.

[M96] R.J. McEliece, "On the BCJR trellis for linear block codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1072-1092, July 1996.

[MRC95] R.J. McEliece, E.R. Rodemich and J.-F. Cheng, "The turbo decision algorithm," *Proc. 33d Allerton Conf. Commun. Control Comput.*, pp. 366-379, Oct. 1995.

[M88] D.J. Muder, "Minimal trellises for block codes," *IEEE Trans. Inform. Theory*, vol. 34, pp. 1049-1053, Sept. 1988.

[N90] R.E. Neapolitan, *Probabilistic Reasoning in Expert Systems*. New York: Wiley, 1990.

[O69] J.K. Omura, "On the Viterbi decoding algorithm," *IEEE Trans. Inform. Theory*, vol. IT-15, pp. 177-179, Jan. 1969.

[P88] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Mateo, CA: Morgan Kaufmann, 1988.

[R89] L. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proc. IEEE*, vol. 77, pp. 257-285, 1989.

[SS96] M. Sipser and D.A. Spielman, "Expander codes," *IEEE Trans. Inform. Theory*, to appear, Nov. 1996.

[S96] D.A. Spielman, "Linear-time encodable and decodable error-correcting codes," *IEEE Trans. Inform. Theory*, to appear, Nov. 1996.

[T81] R.M. Tanner, "A recursive approach to low-complexity codes," *IEEE Trans. Inform. Theory*, vol. IT-27, pp. 533-547, Sept. 1981.

[U71] G. Ungerboeck, "Nonlinear equalization of binary signals in Gaussian noise," *IEEE Trans. Commun. Technol.*, vol. COM-19, pp. 1128-1137, Dec. 1971.

[V67] A.J. Viterbi, "Error bounds for convolutional codes and an asymptotically optimum decoding algorithm," *IEEE Trans. Inform. Theory*, vol. IT-13, pp. 260-269, Apr. 1967.

[WLK95] N. Wiberg, H.-A. Loeliger and R. Kötter, "Codes and iterative decoding on general graphs," *Euro. Trans. Telecommun.*, vol. 6, pp. 513-526, Sept. 1995.

[W96] N. Wiberg, "Codes and decoding on general graphs," Ph.D. dissertation, Dept. Elec. Engg., U. Linköping, Sweden, Apr. 1996.

A Two-phase Auction Algorithm for Highly Asymmetric Assignment Problems

Natalya Cohen
MIT AI Laboratory
nat@ai.mit.edu

Jack Brassil
Bell Laboratories
jtb@bell-labs.com

Abstract

Bertsekas' auction algorithm is an efficient method for solving linear network flow problems, including the classical assignment problem. Empirical studies indicate that the algorithm matches or outperforms its competitors for a large class of problems. The auction algorithm is also well suited for parallel implementation. In this paper we introduce a method to apply the auction algorithm in two phases to solve highly asymmetric assignment problems. In the first phase, the assignment problem is decomposed into subproblems which are solved in parallel; the auction algorithm is used to exclude certain suboptimal assignments from consideration in the second phase. In the second phase, an auction finalizes the optimal assignment. We show that the two-phase algorithm can reduce the theoretical time bound for solving an nxk assignment problem ($n<k$) by a factor of $\sqrt{\frac{k}{n}}$.

1 Introduction

A solution to the *symmetric assignment problem* matches n people with n objects on a one-to-one basis. The benefit for matching a person with an object is given, and the goal is to assign each person a distinct object to maximize the total benefit. The *asymmetric assignment problem* relaxes the restriction that the number of objects must equal the number of people. Here, n people are matched with k objects, with either $n<k$ or $n>k$. The remaining people or objects simply remain unassigned.

Both symmetric and asymmetric assignment problems have many important applications. Both types of problems routinely arise in resource allocation, such as assigning machines to tasks, or personnel to jobs. An unusual military application, reported by Schwartz [11], is the Weapon-Target Assignment problem; the problem is to assign missile interceptors to missiles. In addition to these applications, the assignment problem is of theoretical importance, because the classical linear network flow problem can be reduced to the assignment problem [8]. As a consequence, many algorithms have been introduced to solve the assignment problem; see [10] for examples.

Bertsekas' *auction algorithm* is a novel approach to solving the assignment problem [2] [3] [4] [8]. This is an iterative algorithm reminiscent of an auction, in which people "bid" for objects, and object assignments are made to people willing to pay the highest "price." Object prices eventually stabilize in such a way as to force a one-to-one correspondence between people and objects yielding the maximum total benefit. Hence, the algorithm has a familiar, accessible economic interpretation.

Bertsekas and Castañon [5] have also adapted the auction algorithm to solve the asymmetric assignment problem. In its simplest form, the solution imposes only a slight restriction on the original auction algorithm. Variants of the basic assignment problem, including multiassignment (i.e. more than one object per person) and assignment with similar objects, have also been solved by auction. Moreover, the auction algorithm has been extended to solve network flow problems other than the assignment problem, such as shortest path, max-flow, transportation, and transshipment problems [2] [3].

For a large class of assignment problems the auction algorithm substantially outperforms its competitors, both in theory and in practice [2] [7] [8]. In addition, the algorithm lends itself to parallel implementation. Both synchronous and asynchronous parallel versions of the algorithm have been explored by Bertsekas and Castañon in [6]. The performance of these algorithms on a variety of parallel computers has been studied by other research groups [1] [12] [13]. Many enhancements to the basic algorithm (e.g. the bid adjustment procedure known as ϵ-scaling) have been shown to speedup execution time.

In this paper we present a *two-phase auction algorithm* for large, highly asymmetric assignment problems with $n \ll k$; the case $n \gg k$ is considered in the extended version of this paper. In the first phase, the assignment problem is decomposed into subproblems which are solved in parallel; the auction algorithm is used to exclude certain suboptimal assignments from consideration in the second phase. In the second phase, an auction finalizes the optimal assignment. With the first phase subproblems executed in parallel, the two-phase algorithm can reduce the worst case time required to solve an asymmetric assignment auction algorithm for the case $n \ll k$ by a factor of $\sqrt{\frac{k}{n}}$. In addition, we claim that essentially all of the performance enhancements which can be applied to the basic auction algorithm can also be applied to the two-phase algorithm.

The remainder of this paper is organized as follows. Sections 2 and 3 introduce and analyze the auction algorithm for symmetric and asymmetric assignment problems, respectively. Some of the material found here is well developed in [4], yet is needed to establish a foundation for analyzing the new algorithm. In section 4, we describe the proposed two-phase auction algorithm, and analyze its complexity. We also suggest an approach to overcoming subproblem infeasibility based on introducing artificial arcs (i.e. "dummy" benefits) in a fashion related to the notion of C-scaling [3]. Our conclusions are drawn in section 5.

2 Symmetric Assignment Problem

Suppose we seek to assign each of n persons in a set P to one of n object in a set Q. Associated with each person i is a non-empty set of objects $A(i) \subseteq Q$ to which it can be matched. A non-negative *benefit* $a(i,j)$[1] is given for matching person i with object $j \in A(i)$. An *assignment* S is a set of person-object pairs (i,j) with $j \in A(i)$ for each $(i,j) \in S$; for every person i there can be at most one pair $(i,j) \in S$, and for every object j there can be at most one pair $(i,j) \in S$. A *complete* assignment is one in which every person (and therefore, every object) is assigned. The problem is said to be *feasible* if there exists at least one complete assignment. The solution to the problem is a complete assignment $\{(1,j_1),\ldots,(n,j_n)\}$ with maximum total benefit $\sum_{i=1}^{n} a(i,j_i)$. Note that a solution need not be unique.

[1] We can assume $a(i,j) \geq 0$ without loss of generality; any negative benefit can be made nonnegative by increasing all benefits by the magnitude of the lowest (i.e., the most negative) benefit.

2.1 Auction Algorithm

The auction algorithm proceeds in iterations. Each iteration comprises a bidding phase and an assignment phase. A price vector p specifying object prices and a partial (i.e., not complete) assignment S are kept throughout the algorithm. The algorithm terminates when the assignment is complete.

At the beginning of each iteration, S and p satisfy the *ϵ-complementary slackness* (ϵ-CS) condition,

$$a(i, j_i) - p_{j_i} \geq \max_{j \in A(i)} \{a(i, j) - p_j\} - \epsilon \tag{1}$$

for all pairs $(i, j_i) \in S$. If all persons are assigned, the algorithm terminates. Otherwise, during the bidding phase, each unassigned person i finds each of the following: an object j_i with the maximal value

$$j_i = arg \max_{j \in A(i)} \{a(i, j) - p_j\}, \tag{2}$$

the corresponding best object value v_i,

$$v_i = \max_{j \in A(i)} \{a(i, j) - p_j\}, \tag{3}$$

and the second best object value w_i

$$w_i = \max_{j \in A(i), j \neq j_i} \{a(i, j) - p_j\}. \tag{4}$$

Person i submits a bid for object j_i at a price given by

$$p_{j_i} := p_{j_i} + v_i - w_i + \epsilon. \tag{5}$$

After bids are submitted, the assignment phase begins. Each object receiving a bid identifies its maximum bidder i. The pair (i, j_i) is then added to S, and if j_i was assigned to another person i' at the start of the iteration, the pair (i', j_i) is removed from S. Note that S and p at the end of the iteration still satisfy the ϵ-CS condition (1).

It is shown in [4] that if at least one complete assignment exists, the auction algorithm terminates. Furthermore, a complete assignment and final price vector p which satisfies ϵ-CS is within $n\epsilon$ of being optimal. It follows that if $\epsilon < \frac{1}{n}$, the benefits $a(i, j)$ are all integer-valued, and the initial p and S satisfy ϵ-CS, then the assignment obtained upon termination of the auction algorithm is optimal. In the absence of any other information, the empty assignment $S = \emptyset$ and the zero price vector p can be used for the initial S and p. Finally, Bertsekas has identified two versions of the basic algorithm. In the *Gauss-Seidel* version, prices are adjusted after each person submits a bid; in the *Jacobi* version, prices are adjusted after all unassigned persons submit a bid.

2.2 Complexity Analysis

There are $n!$ possible solutions to the symmetric assignment problem if each person can be matched to any object. Yet the auction algorithm is shown in [8] to have the worst-case running time of $O(A(n + n\frac{C}{\epsilon}))$, where

$$A = \sum_{i=1}^{n} |A(i)| \tag{6}$$

is the total number of possible matchings between all people and objects, and

$$C = \max_{\substack{1 \le i \le n \\ j \in A(i)}} |a(i,j)| \tag{7}$$

is the maximum absolute object benefit. To simplify analysis, let us assume that $A = \Theta(n^2)$, and that $C = \Omega(\epsilon)$. Then, the worst-case running time is given by

$$O(n^3 \frac{C}{\epsilon}). \tag{8}$$

To understand this result, suppose the maximum amount by which the price of any object changes during the course of the algorithm is β. It can be seen from equations (3)-(5) that the price p_j of any object j does not decrease during the algorithm's execution; furthermore, whenever p_j increases, it increases by at least ϵ. It follows that the maximum number of times an object's price changes during the course of the algorithm is $\frac{\beta}{\epsilon}$.

Note further that each iteration of the algorithm (i.e. Gauss-Seidel version) results in a price change of exactly one object. For n objects, we conclude that the maximum number of iterations is $n\frac{\beta}{\epsilon}$. Examination of benefits for a single unassigned person i takes time $O(|A(i)|) = O(n)$. The conclusion is that the entire algorithm takes time $O(n^2\frac{\beta}{\epsilon})$. Finally, a bound on any object's price change is given by $\beta = O(nC)$ ([4], p. 234). Combining these results yields the time bound given in equation (8).

Auction algorithm execution time can be further reduced by the use of *ϵ-scaling*, as discussed in [7] and [8]. The basic idea behind ϵ-scaling is to have initial iterations start with a large value for ϵ and successively reduce ϵ down to a value that is less than $\frac{1}{n}$. Each bidding round provides high initial prices for the next round, thus accelerating the time to achieve equilibrium prices. The time bound for this version of the algorithm is proven in [8] to be $O(n^3 log(nC))$.

3 Asymmetric Assignment Problem

In the asymmetric assignment problem, the set Q contains k objects, with $k \neq n$. For the case $n < k$, the problem specification does not change. The assignment S is still considered complete if every person is assigned (in this case, however, $k - n$ objects remain unassigned), and the solution to the problem is a complete assignment $\{(1, j_1), \ldots, (n, j_n)\}$ yielding the maximum total benefit $\sum_{i=1}^{n} a(i, j_i)$.

There are three alternative approaches to adapting the auction algorithm to solve assignment problems with fewer people than objects. The most straightforward approach is to first convert the asymmetric assignment problem into a symmetric assignment problem, and then apply the auction algorithm. In order to convert from an asymmetric to a symmetric problem, we create $k-n$ "dummy" people. Each dummy person can be matched with any object, and for any dummy person i and any object $j \in Q$, we set $a(i,j) = 0$ (i.e. the minimum benefit possible). After the algorithm's termination, the dummy people are matched to objects that no real person wanted. Dummy person-object pairs are discarded, and the assignment that remains is the solution to the asymmetric assignment problem.

The approach described above carries the overhead of iterations involving dummy people, only to discard them at the algorithm's termination. Specifically, the worst-case running time is the same as for the kxk (symmetric) problem, i.e. $O(k^3\frac{C}{\epsilon})$.

The second approach to solving the asymmetric problem is based on the observation that the dummy persons introduced above are unnecessary. We first *constrain all initial prices to zero*, and start with an empty assignment. The auction algorithm is run only *until all n people are assigned* (with $k-n$ objects remaining unassigned). The final assignment $S = \{(1, j_1), \ldots, (n, j_n)\}$ obtained by this method solves the asymmetric assignment problem. This result is proved in the extended paper.

This method of solving the asymmetric assignment problem eliminates the need for dummy people, decreasing the worst-case running time. The complexity analysis parallels that for the symmetric case. It is still the case that object price increases are bounded by $\beta = O(nC)$. For any object, the maximum number of times the object's price changes is $\frac{\beta}{\epsilon}$. Furthermore, the objects that are unassigned at the end of the algorithm stay at their initial price throughout the course of the algorithm. Thus, only n out of k objects ever change their price; the maximum number of iterations is therefore $n\frac{\beta}{\epsilon}$. With each iteration taking time $O(|A(i)|) = O(k)$ (for some unassigned person i), the entire algorithm takes time $O(nk\frac{\beta}{\epsilon})$, or

$$O(n^2 k \frac{C}{\epsilon}). \tag{9}$$

Note that, in the case of $n = k$, the quantities (8) and (9) are equal, as expected.

The requirement of zero initial prices imposed by the above approach is unfortunately incompatible with ϵ-scaling. To get around this problem, Bertsekas and Castañon developed a third approach to solving the asymmetric assignment problem, the *forward/reverse auction algorithm*, which allows for ϵ-scaling and yields better performance [5]. In essence, the forward/reverse auction algorithm involves both people bidding for objects, and objects competing for people by offering discounts.

To simplify the complexity analysis for the two-phase auction algorithm introduced in the next section, we assume in the rest of the paper that the asymmetric assignment problems ($n < k$) are solved by imposing the requirement of zero initial prices and running the auction algorithm until all n people are assigned (method 2 described above); the worst-case running time for this approach is given by equation (9) above. It should be understood, however, that in practice both the ϵ-scaling and the forward/reverse auction algorithm techniques can be used to enhance the performance of the proposed two-phase algorithm.

4 Highly Asymmetric Assignment Problem

Large optimization problems often permit solution by algorithms capable of exploiting their special structure [9]. We now present a two-phase auction algorithm for highly asymmetric assignment problems with $n \ll k$. The two-phase algorithm decomposes the initial problem into smaller asymmetric subproblems to be solved. The two-phase algorithm uses the auction algorithm for asymmetric assignment problems with $n < k$ (described in subsection 3) as a building block; in this section, we refer to it as simply "the auction algorithm."

4.1 Two-phase Auction Algorithm: $n \ll k$

Suppose we have an assignment problem for a set P of n people and a set Q of k objects, with $n \ll k$. To simplify exposition we will assume that k is an integer multiple of n,

and k is an integer multiple of d, a parameter to be introduced shortly. The two-phase auction algorithm proceeds as follows.

In phase 1, the set of objects Q is partitioned into d disjoint subsets $Q_\alpha : \alpha = 1, \ldots, d$, each containing m objects ($m > n$) and satisfying

$$Q = \bigcup_{\alpha=1}^{d} Q_\alpha.$$

The auction algorithm is then run in parallel on the set of all persons P and each of the subsets of objects Q_α. Let us assume for the moment that for each α, the assignment problem for the sets P and Q_α is feasible. We will address the issue of infeasible subproblems at the end of this section.

For each set Q_α, the result of phase 1 is the assignment $\{(1, v_1), \ldots, (n, v_n)\}$ yielding the maximum total benefit $b_\alpha = \sum_{i=1}^{n} a(i, v_i)$. $V_\alpha = \{v_1, \ldots, v_n\}$ is the set of n objects that are the "winners" of phase 1 of the assignment problem for the set Q_α (e.g. object v_1 is won by person 1, and so on).

For phase 2 of the algorithm, we collect the set of n winners from each of the d subsets of Q (nd objects total), and run the auction algorithm again, on the set of nd objects and the same set P of people. The result is the assignment $\{(1, w_1), \ldots, (n, w_n)\}$, yielding the maximum total benefit $b = \sum_{i=1}^{n} a(i, w_i)$. $W = \{w_1, \ldots, w_n\}$ is the set of phase 2 winners.

The proof of correctness of this algorithm is found in the extended paper. There we prove that the set of objects W maximizes the total benefit for the assignment problem on sets P and Q. That is, if the auction algorithm were run on the sets P and Q, the resulting benefit would equal that corresponding to the assignment of each person i and object w_i. Thus, the two algorithms each yield a complete assignment with identical benefit, and so the two-phase algorithm solves the assignment problem.

4.2 Complexity Analysis

To analyze the complexity of the two-phase algorithm we make use of (9), the formula for the worst-case running time of the auction algorithm.

In phase 1, each auction algorithm is performed in parallel on n people and m objects, with $m = \frac{k}{d}$. Thus, the running time for phase 1 of the algorithm is $O(n^2 m \frac{C}{\epsilon}) = O(\frac{n^2 k}{d} \frac{C}{\epsilon})$. In phase 2, the auction algorithm is performed on n people and nd objects, so the running time is $O(n^3 d \frac{C}{\epsilon})$. The total running time is thus $O((\frac{n^2 k}{d} + n^3 d) \frac{C}{\epsilon})$.

The two-phase algorithm performs best when the quantity $\frac{n^2 k}{d} + n^3 d$ is minimized. For a fixed n and k, the running time is minimized when

$$d = \sqrt{\frac{k}{n}}. \tag{10}$$

Using (10) and the relation $k = md$, we obtain

$$k = md = nd^2 \tag{11}$$

$$m = nd. \tag{12}$$

Eq. 12 reveals that the two-phase algorithm performs best when the number of objects participating in phase 1 of the algorithm (for each Q_α) is equal to the number of objects participating in phase 2. For example, if we start with 250 objects and 10 people, the

optimal way to split objects is in 5 groups of 50 objects each; the number of objects participating in phase 2 of the algorithm will again be 50.

For the case when (10) holds, the worst-case running time is

$$O((\frac{n^2k}{\sqrt{\frac{k}{n}}} + n^3\sqrt{\frac{k}{n}})\frac{C}{\epsilon}),$$

which reduces to

$$O(n^{\frac{5}{2}}k^{\frac{1}{2}}\frac{C}{\epsilon}). \tag{13}$$

Comparing (9) and (13), we find that the two-phase algorithm has lower time complexity than the auction algorithm whenever $n < k$.

4.3 Feasibility of Subproblems

To this point we have assumed that, in phase 1 of the two-phase algorithm, the assignment problem for set P and and each Q_α is feasible. We now turn to the important case in which this assumption does not hold. It is easy to see how this might happen. As a pathological example, it may turn out that some person $i \in P$ is not interested in any objects in the set Q_α for a particular α; since i cannot be assigned to any object in Q_α, the assignment problem for sets P and Q_α is infeasible.

For the two-phase auction algorithm to run faster than the auction algorithm, we need to either ensure subproblem feasibility, or quickly detect infeasibility. In this section we introduce an approach to rapidly detecting *potentially* infeasible subproblems, and an approach to handling them. But prior to introducing this approach, we will first seek to determine the frequency of subproblem infeasibility.

We will next show that even relatively sparse benefit matrices produce, on average, an overwhelming fraction of subproblems which are feasible. Consider an nxk assignment problem with $n < k$. Suppose that for each person i the set of objects of interest $A(i)$ is constructed randomly. Assume that, for each person, each object is chosen to belong to the interest set independently with probability p. Hence, for an nxk benefit matrix, we expect the number of objects of interest to each person to be pk. Hence, we can say that the parameter p is a measure of benefit matrix occupancy.

Suppose that in phase 1 we decompose the problem into exactly d nxm subproblems. It is easy to show that the probability that each nxm subproblem matrix is feasible is given by

$$p(n,m) = \prod_{l=0}^{n-1}(1-(1-p)^{m-l}). \tag{14}$$

A subset of the d subproblems will be feasible, and each of these will contribute n objects to the second phase auction. Suppose the remaining, infeasible subproblems will be handled in the following fashion. When infeasibility is detected, the subproblem is aborted; all m objects are passed to the second phase auction. Then the expected number of people entering the second phase auction is given by

$$ndp(n,m) + k(1-p(n,m)). \tag{15}$$

We now turn to the method of detecting subproblem infeasibility, which we hinted at above. One way to detect infeasible problems in the context of an auction is described by Bertsekas in [3]. The method consists of converting the problem to a feasible problem by expanding the set $A(i)$ of desirable objects for each person i. The value of the benefits $a(i,j)$ added by this expansion (let us call them "dummy" benefits) should be very small, so that none of the added person-object pairs participate in an optimal assignment unless the problem is infeasible. In particular, Bertsekas shows that if the original problem is feasible, no person-object pair with a dummy benefit (let us call it a "dummy" person-object pair) will participate in the optimal assignment, provided that each dummy benefit satisfies

$$a(i,j) < -(n-1)C. \tag{16}$$

We can use this mechanism to ensure the feasibility of subproblems created by the two-phase algorithm. In particular, at the start of the algorithm, we can expand the set $A(i)$ for each person i to include all the objects in Q, setting all dummy benefits to the value $-(n-1)C-1$.[2] After phase 1 of the algorithm, dummy person-object pairs may very well be advanced to phase 2, even if the assignment problem for sets P and Q is feasible; however, unless the problem for sets P and Q is infeasible, no dummy pair will participate in the final assignment for phase 2.

Note also that, if the feasibility of the entire problem is a given, then there is no need to adapt the dummy benefits in phase 2 of the algorithm, since we are ensured that the problem in phase 2 is feasible. Thus, we only have to make sure that each subproblem in phase 1 is feasible. To guarantee this, it is not necessary to expand the set $A(i)$ for each person i to include all the objects participating in the subproblem. For example, in the case of $n \ll k$, we might simply make sure that each person i is interested in at least one object, say object i; if this is not the case, we then expand the sets $A(i)$ as needed to make it so. This guarantees the existence of at least one complete assignment for the subproblem—namely, the one in which each person i is assigned to the object i—and thus, the subproblem's feasibility.

The problem with the above method is that the addition of dummy benefits raises the cost range of the problem, C, by a factor of $O(n)$. This results in an increase of the running time of the algorithm, which is proportional to C (see equation (13)). In the context of ϵ-scaling, this necessitates a much larger number of scaling phases, which are wasted if the problem is feasible. To remedy the situation somewhat, Bertsekas suggests a form of cost scaling (i.e. $C - scaling$) [3]; start off with dummy benefits on the order of $-C$, and then scale them downward toward $-(n-1)C$ (e.g. as part of the ϵ-scaling procedure) if any dummy person-object pairs persist in the assignments obtained at the end of each scaling phase.

If the assignment problem is moderately dense (and thus the number of infeasible subproblems is small), the above modification to the two-phase algorithm should not affect the algorithm's performance substantially. However, for sparse problems with a great number of infeasible subproblems, the overall performance may suffer significantly.

Another approach to handling infeasible subproblems tries to make a *conservative estimate* of the subproblem's infeasibility. If the subproblem in phase 1 is determined to be feasible, the assignment it produces is trusted, and the winners of that subproblem are

[2]To ensure that the condition $a(i,j) \geq 0$ is satisfied for all i and j, we could instead set each dummy benefit to 0, and raise all the real benefits by $(n-1)C+1$.

advanced into phase 2, as before. If, on the other hand, it cannot be rapidly determined whether or not the subproblem is feasible, all objects in the subproblem are advanced into phase 2, so as to make sure no eventual winners are omitted from phase 2.

More specifically, suppose we still convert each subproblem α of phase 1 into a feasible subproblem as above (e.g. by expanding the set $A(i)$ to include object i if not in the set), but instead of making the dummy benefits very small, we set them to -1, a value just a little smaller than any real benefit. Now, suppose the auction algorithm is run on the modified subproblem. If the final assignment contains no dummy person-object pairs, then we know that the subproblem was feasible to begin with (otherwise, no complete assignment without dummy pairs would be possible); we can thus advance the subproblem's winners into phase 2, as before. Otherwise, we have no guarantees about the subproblem's feasibility. If the subproblem is indeed feasible, but our final assignment nonetheless contains dummy pairs, we cannot be sure that our assignment does not exclude some eventual winners of the assignment problem for sets P and Q. Thus, to guarantee that no eventual winners are omitted from phase 2, we advance the entire subproblem into the second phase. In the case of $n \ll k$, this means that the entire set Q_α of objects participates in phase 2.

Note that this method does not decrease the performance of phase 1 of the two-phase algorithm. In fact, this approach can actually speedup the initial phase. Suppose that interest set of each person is expanded to include a dummy person-object pair with associated benefit -1. In this case, at every iteration of the auction algorithm, each unassigned person has available an unassigned object at this price. As the prices of real objects are "bid up," a dummy object can become the most valued object to each person. Indeed, the addition of this "low cost" object reduces the bound on object price increases to $\beta = O(C)$. Hence, phase 1 subproblems are solved (or determined infeasible) in worst case time which is

$$O(nm\frac{C}{\epsilon}). \tag{17}$$

Of course, first phase problems which can not be solved in this time expand the size of the second phase auction. However, we showed in (15) that for all but the most sparse randomly constructed benefit matrix, the likelihood of subproblem feasibility is quite low.

5 Conclusion

In this paper we presented a two-phase auction algorithm for highly asymmetric assignment problems. The first phase of the algorithm is used to "filter" certain suboptimal assignments from consideration in the second phase. The second phase realizes an optimal assignment. For the case where all first phase subproblems are feasible, we proved algorithm correctness and derived bounds on the algorithm running time. We also presented an approach to handling subproblem infeasibility which offers a promising compromise – the approach solves those feasible subproblems which can be solved rapidly, yet quickly abandons those subproblems which can not.

The worst-case running time bound (13), obtained on the assumption that all the subproblems are feasible, represents an improvement over the time bound (9) for the auction algorithm. Even without the assumption of feasibility of subproblems, the two-phase algorithm shows potential for improved performance, particularly for dense problems.

Much research remains to be done on the two-phase algorithm, both on the theoretical and the practical fronts. On the theoretical side, note that we have considered only an arbitrary partitioning of the set Q of objects (for the case $n \ll k$) in phase 1 of the algorithm. Given some knowledge of problem structure, there might exist more clever ways to partition the problem which could further reduce the number of infeasible phase 1 subproblems. In addition, the "all or nothing" abandonment of phase 1 subproblems seems suboptimal. Our experiments show that, in many cases, useful filtering of suboptimal assignments does occur (for those phase 1 subproblems which are not solved) but are nonetheless abandoned.

On the practical side, the algorithm's utility must be further demonstrated by performing extensive numerical experiments on representative assignment problems, and comparing the results with the performance of other methods (in particular, the auction algorithm). In addition, various enhancements of the two-phase algorithm suggested in this paper have to be implemented, and their utility evaluated.

References

[1] M. M. Amini, Vectorization of an Auction Algorithm for Linear Cost Assignment Problem, *Computers & Industrial Engineering*, Vol. 26, No. 1, pp. 141-149, January 1994.

[2] Dimitri P. Bertsekas, The Auction Algorithm for Assignment and Other Network Flow Problems: A Tutorial, *Interfaces*, Vol. 20, No. 4, pp. 133-149, July-August 1990.

[3] Dimitri P. Bertsekas, Auction Algorithm for Network Flow Problems: A Tutorial Introduction, *Computational Optimization and Applications*, Vol. 1, pp. 277-297, 1992.

[4] Dimitri P. Bertsekas, *Linear Network Optimization: Algorithms and Codes*, MIT Press, Cambridge, Massachusetts, 1991.

[5] Dimitri P. Bertsekas, David A. Castañon, A Forward/Reverse Auction Algorithm for Asymmetric Assignment Problems, *Computational Optimization and Applications*, Vol. 1, pp. 277-297, 1992.

[6] Dimitri P. Bertsekas, David A. Castañon, Parallel Synchronous and Asynchronous Implementations of the Auction Algorithm, *Parallel Computing*, Vol. 7, No. 6-7, pp. 707-732, September 1991.

[7] Dimitri P. Bertsekas, Jonathan Eckstein, Dual Coordinate Step Methods for Linear Network Flow Problems, *Mathematical Programming*, Series B, Vol. 42, No. 2, pp. 203-243, 1988.

[8] Dimitri P. Bertsekas, John N. Tsitsiklis, *Parallel and Distributed Computation: Numerical Methods*, Prentice Hall, Englewood Cliffs, New Jersey, 1989.

[9] L. L. Lasdon, *Optimization Theory for Large Systems*, MacMillan Co., NY, 1970.

[10] C. H. Papadimitriou, K. Steglitz, *Combinatorial Optimization: Algorithms and Complexity*, Prentice-Hall, Englewood Cliffs, NJ, 1982.

[11] B. L. Schwartz, A Computational Analysis of the Auction Algorithm, *European Journal of Operational Research*, Vol. 74, No. 1, pp. 161-169, April 7, 1994.

[12] Joel M. Wein, Stavros Zenios, Massively Parallel Auction Algorithms for the Assignment Problem, The 3rd Symposium on the *Frontiers of Massively Parallel Computation*, ed. J. JaJa, IEEE Computer Society Press, 1990.

[13] Hossam A. Zaki, A Comparison of Two Algorithms for the Assignment Problem, *Computational Optimization and Applications*, Vol. 4, No. 1, pp. 23-45, 1995.

Fairness in Cellular Mobile Networks*

Saswati Sarkar and *Kumar N. Sivarajan* †
Electrical Communication Engineering Department
Indian Institute of Science
Bangalore 560 012, INDIA
email addresses: swati@ece.iisc.ernet.in, kumar@ece.iisc.ernet.in

Abstract

Channel allocation algorithms for channelized cellular systems are discussed from a new perspective, viz., fairness of allocation. The concepts of relative and absolute fairness are introduced and discussed. It will be shown that under certain reasonable assumptions, there exists an absolute (maxmin) fair carried traffic intensity vector (a vector describing the traffic carried in the cells of the system). We also show that this vector is unique. We describe some properties of the maxmin fair carried traffic intensity vector in an asymptotic limit where the traffic and the number of channels are scaled together. For each traffic pattern, we determine a fixed channel allocation which attains this maxmin fair carried traffic intensity vector independent of the value of the offered traffic, in the same asymptotic limit. Finally we discuss a tradeoff between being maxmin fair and trying to maximize revenue. We conclude this paper by discussing some possible extensions of our work.

1 Introduction and Summary

In a cellular system, the coverage area is logically divided into cells. Each cell has a cell site or a base station. The communication from the mobile user is directed to a central switching office by the base station. The central switching office directs this communication to the destination. Depending on the mode of multiple access used by the mobile customers, cellular systems can be broadly classified into *channelized* and *nonchannelized* systems. In a channelized cellular system the multiple access is TDMA or FDMA or a combination of both. The term *channel* refers to a time slot in TDMA, a frequency slot in FDMA and a combination of both in TDMA/FDMA systems like GSM. Calls arrive and depart at random times in the cells of the system and a *channel assignment algorithm* must assign a channel to each call for its duration, while obeying certain channel reuse constraints. These reuse constraints can be modelled by a hypergraph[1], as explained in [2]. This paper will deal entirely with channelized systems. The cellular network operator whose task it is to choose an appropriate channel assignment algorithm is usually interested in maximizing his revenue and this amounts to maximization of the total traffic

*This research was supported by a grant from Bell Northern Research.
†Corresponding author

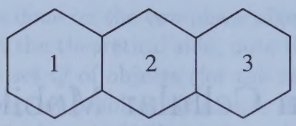


Figure 1: A 3-cell system

carried in the system, if we assume calls in all cells are charged at the same rate ($ per unit time). In addition, in order to keep his customers satisfied, the operator has to provide them a minimum grade of service (maximum blocking probability). However, many channel assignment algorithms provide unequal grades of service in the various cells of the system, and are thus "unfair." This paper attempts to study the concept of fairness in cellular networks. First we shall motivate the study of fairness with an example.

Example 1.1 Consider the 3-cell system shown in figure 1 and assume that a total of 80 channels are available. Adjacent cells are forbidden from using the same channel, but cells 1 and 3 can reuse the same channel. Consider two fixed channel algorithms, FCAA1 and FCAA2. FCAA1 allocates 40 channels to each cell and FCAA2 allocates 42 channels to cells 1 and 3 and 38 channels to cell 2. Let the offered traffic in each cell be 35 Erlangs. Let us assume that there are no handovers and no intercell calls. We also assume that neither algorithm allows any queuing of call requests. If there is a free channel in the cell in which a call is requested, the call request is honored; otherwise it is blocked. FCAA1 achieves a blocking probability of 5.4% in each cell and the total carried traffic is 99.3 Erlangs. For FCAA2 the blocking probability is 3.6% in cells 1 and 3 and 7.8% in cell 2. The total carried traffic is 99.8 Erlangs, slightly better than that of FCAA1. FCAA1 can be said to be "fairer" than FCAA2 since it treats the individual cells more "equally." It is reasonable to assume that in this case a network operator will prefer FCAA1 to FCAA2 even though his total carried traffic is slightly reduced. Otherwise, he will risk the desertion of customers in cell 2 for a competitor operator. Thus the problem of ensuring some level of fairness in channel allocation becomes an interesting one.

We remark that fairness does not necessarily imply "equal blocking probability." Various definitions of fairness are possible but the most appropriate one appears to be the notion of "maxmin fairness." A similar notion of fairness is widely used in the context of flow control [4]. Roughly speaking, maxmin fairness minimizes the overall blocking probability without decreasing the blocking probability in any cell at the expense of other cells which are already worse off. More precise definitions follow in later sections.

We make the following assumptions regarding the cellular system. The system consists of N cells and the underlying offered traffic model is independent from cell to cell; in particular, we ignore the effect of call handovers and intercell calls. The call requests form a Poisson process and the call duration is exponentially distributed. The N cells share a common set of n channels. If A_i denotes the offered traffic in cell i (the expected number of calls that would be in progress in cell i if all call requests could be honored), then A_i/n is the *offered traffic intensity* in cell i. The offered traffic intensity in the system, r, is the aggregate of the offered traffic intensities in the cells. The ratio $p_i = A_i / \sum_{i=1}^{N} A_i$ represents the fraction of the total offered traffic in cell i and the vector $\tilde{p} = (p_1, p_2 \ldots p_N)$ is the *traffic pattern*. The carried traffic intensity in cell i, x_i, is the carried traffic (expected number of calls in progress) in cell i per available channel in the system. We call the vector $\tilde{x} = (x_1, x_2 \ldots, x_N)$ the *carried traffic intensity vector*. When a call

request arrives in a cell, the channel assignment algorithm either assigns it a channel for its entire duration, or blocks it. (A blocked call disappears from the system). The decision regarding blocking or honoring a call request is taken by the channel assignment algorithm based entirely on the current value of a suitably defined *state vector*. Further we assume that the state vector assumes a finite, or countably infinite, set of values. An example of a state vector is a list that specifies the calls in progress in each cell of the system and the channels assigned to them. The assignment of channels to calls at any stage must satisfy certain reuse constraints which essentially specify the sets of cells that may simultaneously use the same channel. Each possible channel assignment algorithm corresponds to one and only one carried traffic intensity vector under fixed r, p and n. We denote the set of achievable carried traffic intensity vectors under fixed r, p and n, by $F(r,\ p,\ n)$ or simply F. The blocking probability in cell i is $b_i = 1 - x_i/p_i r$ and the call acceptance probability in cell i is given by $1 - b_i = x_i/p_i r = c_i/r$, where $c_i = x_i/p_i$. We consider only systems in which $p_i > 0$ for all i[‡]. The carried traffic, blocking probability and call acceptance probability depend on the channel assignment algorithm used and we shall denote this by an appropriate superscript. E.g., $\tilde{x}^A$ is the carried traffic intensity vector under channel assignment algorithm A.

We will proceed as follows. In Section 2 we formally define the concept of relative fairness and present a simple necessary and sufficient condition for relative fairness. In Section 3 we introduce the notion of absolute fairness and prove its existence and uniqueness. In Section 4 we deal with fairness in the asymptotic limit, i.e., when the number of channels is arbitrarily large. We also discuss the computation of the absolutely fair carried traffic intensity vector in the asymptotic limit and give the corresponding channel allocation strategy to attain it. In Section 5 we discuss the loss of revenue brought about by fairness and discuss a tradeoff between being fair and maximizing the revenue. We conclude this paper by discussing how our work can be extended.

The theorems will be stated without proof. The omitted proofs will be presented in a full version of the paper. However we will give outlines of the proofs of some of our important results.

2 Relative fairness

Informally, a channel assignment algorithm A is fairer than another channel assignment algorithm B, for given r, p and n, if for every cell i whose blocking probability is decreased by B compared to A, there is some other cell j whose blocking probability was already no less than that of i under A and has been increased further by B. A more formal definition of relative fairness is as follows.

A channel assignment algorithm A is *fairer* than another channel assignment algorithm B, under the same r, p and n if

- $\tilde{x}^A \neq \tilde{x}^B$ and
- if there exists an i such that $b_i^A > b_i^B$, then there exists a j such that $b_j^A \geq b_i^A$ and $b_j^B > b_j^A$. Equivalently, if there exists an i such that $c_i^A < c_i^B$, then there exists a j such that $c_j^A \leq c_i^A$ and $c_j^B < c_j^A$.

Since a channel assignment algorithm corresponds to a unique carried traffic intensity vector for fixed r, p and n, we will usually speak of the fairness of carried traffic intensity

[‡]This means we eliminate from the system cells with no offered traffic.

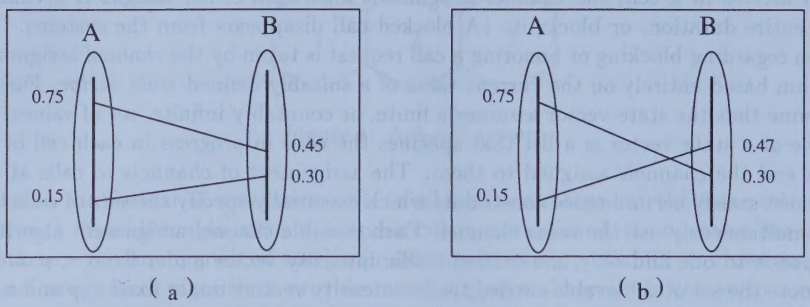


Figure 2: Scale diagram: B is fairer than A in (a). Neither A nor B is fairer than the other in (b). $N = 2$. The numbers are the values of the corresponding c_is.

vectors rather than the fairness of channel assignment algorithms. Note that given any one of the vectors, $\tilde{x}$, $\tilde{b} = (b_1, \ldots, b_N)$ and $\tilde{c} = (c_1, \ldots, c_N)$, we can determine the other two uniquely (since r and $\tilde{p}$ are fixed). Thus we may also equivalently consider the fairness of the vectors $\tilde{b}$ or $\tilde{c}$ instead of $\tilde{x}$.

Figure 2 (a)[§] shows a pair of vectors, $\tilde{c}^A$ and $\tilde{c}^B$, where $\tilde{c}^B$ is fairer than $\tilde{c}^A$.

Theorem 1 *A channel assignment algorithm A is fairer than another channel assignment algorithm B if and only if there exists an $A_{min} \in S \subseteq U = \{1, 2, \ldots N\}$, such that $c^A_{A_{min}} = \min_{i \in S} c_i^A$ and $c^A_{A_{min}} > c^B_{A_{min}}$, where $S = \{i : c_i^A \neq c_i^B\}$.*

The proof of this theorem follows from the definition of relative fairness.

Remarks: In other words, this theorem states that the necessary and sufficient condition for fairness of a channel assignment algorithm A over another B is that the least call acceptance probability (over all cells) under A is strictly greater than the corresponding call acceptance probability in B (ignoring cells with equal call acceptance probabilities).

3 Absolute Fairness

A channel assignment algorithm is *absolutely fair* or *maxmin fair* if it is fairer than any other channel assignment algorithm which achieves a different carried traffic intensity vector under the same r, p and n. We shall refer to the carried traffic intensity vector corresponding to the maxmin fair channel assignment algorithm as the MMF (MaxMin Fair) carried traffic intensity vector, or simply the MMF. Informally, a carried traffic intensity vector is the MMF if the blocking (probability) in any cell under it cannot be decreased without increasing the blocking in any cell already experiencing greater or equal blocking.

It is not obvious that every set of carried traffic intensity vectors, F, has an MMF. Consider the following examples.

Example 3.1 Let F consist of only the carried traffic intensity vectors corresponding to the vectors $\tilde{c}^A$ and $\tilde{c}^B$ shown in figure $2(b)$. Neither A nor B is fairer than the other. Hence no MMF exists in F in this case.

[§]Scale diagram is a schematic representation of vectors. The components are represented as points on the real line. Corresponding components of vectors are joined.

Example 3.2 Let $F = R^N$. (R^N is the set of $N-$dimensional vectors with real components.) For any vector $\tilde{x}^V \in F$, there exists another vector, $\tilde{x}^C \in F$, each of whose components is greater than the corresponding one in $\tilde{x}^V$. Thus $\tilde{x}^V$ is not fairer than $\tilde{x}^C$ and hence is not the MMF. The same observation applies for $F = \{\tilde{x} = (x_1, x_2, \ldots, x_N) : 0 \leq x_i < 1\}$.

In view of the above examples, it becomes necessary to prove the existence of an MMF. In this section we will first show that any set of vectors with certain properties has an MMF¶. Next we discuss whether the set of achievable carried traffic intensity vectors of a cellular system has these properties. We end this section with the result that the MMF is unique.

First we define the terms closed, bounded, convex, and coordinate convex which we will use subsequently. Unless otherwise mentioned $U = \{1, 2, \ldots, N\}$.

Closed: Consider the metric space (R^N, d). A subset of R^N, F, is *closed* if it meets the following criterion: if any $\tilde{V}$ which has the property that for any $\epsilon > 0$, there exists a vector $\tilde{V}' \in F$, such that $d(\tilde{V}, \tilde{V}') < \epsilon$, belongs to F, then F is closed. We shall use the Euclidean metric as d. However if any subset of R^N is closed (bounded) w.r.t. any norm induced metric, it is also closed (bounded), w.r.t. any other norm induced metric. (Euclidean metric is a norm induced metric) [5]. So all our results apply even if the metric is any other norm induced metric.

Bounded: A subset F of R^N is bounded if for any $\tilde{x}^V \in F$, $|x_i^V| \leq l$,where l is a finite real number, for all $i \in U$.

Convex: A subset of R^N, F, is convex, if a convex combination of any two vectors in F is in F, i.e., if $\tilde{x}^A, \tilde{x}^B \in F$, then $\tilde{x}^C = \alpha\tilde{x}^A + (1-\alpha)\tilde{x}^B$ is in F, for any $\alpha \in (0,1)$.

Coordinate Convex : A subset of R_+^N (R_+^N is the set of all N-dimensional vectors with real nonnegative components), F, is said to be coordinate convex if for each vector $\tilde{x} = (x_1, x_2, \ldots, x_N)$ in F, $\tilde{x}' = (x_1, x_2, \ldots, \alpha x_i, \ldots, x_N)$, belongs to F for all α satisfying $0 \leq \alpha < 1$ and for all $i \in U$.

Theorem 2 *If P is a nonempty, coordinate convex, convex, closed, bounded subset of R^N, P has an MMF.*

We only give a construction for a vector that can be proved to be maxmin fair using Theorem 1.

Construction: Start from the all-zero vector $c_i = 0$, $\forall i \in U$.
Always find the largest subset $I_l \subseteq U$ such that c_i's, $i \in I_l$, can be increased equally without decreasing any c_i. Increase the c_i's of this subset equally without decreasing any c_i, till it is no longer possible to do so.
The procedure terminates when the largest subset which can be increased equally without decreasing any c_i is empty, i.e., no c_i can be increased without decreasing some other c_i.

Informally the procedure goes as follows: Start from the vector $(0, 0 \ldots, 0)$. Suppose the largest subset which can be increased is $\{1, 2, \ldots, k\}$, where $1 \leq k \leq N$. Increase $c_1, \ldots, c_k$ equally till a vector $(p_1\alpha, \ldots, p_k\alpha, 0, \ldots, 0)$ is reached and $\alpha \geq 0$ is such that $(p_1(\alpha+\epsilon), \ldots, p_k(\alpha+\epsilon), 0, \ldots, 0)$ does not exist in P for any $\epsilon > 0$. Again increase the largest possible subset equally without decreasing any component. The largest possible subset will be a proper subset of $\{1, \ldots, k\}$. Let it be $\{1, \ldots, m\}$, $m < k$. The increase of

¶The definition of fairness in any set of N-dimensional vectors remains the same as that for carried traffic intensity vectors. $c_i^V = x_i^V/p_i$, where x_i^V is the ith component of the vector $\tilde{V}$. $\tilde{p} = (p_1, p_2, \ldots, p_N)$ is the same for the entire set. $p_i > 0$, for all $i \in U$.

this subset continues till $(p_1\beta, \ldots, p_m\beta, p_{m+1}\alpha, \ldots, p_k\alpha, 0, \ldots, 0)$ is reached and $(p_1(\beta + \epsilon), \ldots, p_m(\beta + \epsilon), p_{m+1}\alpha, \ldots, p_k\alpha, 0, \ldots, 0)$ is not in P for any $\epsilon > 0$. The procedure terminates when no element can be increased without decreasing some other(s).

We discuss the validity of the assumptions that the set of carried traffic intensity vectors is nonempty, closed, bounded, convex and coordinate convex.

Nonempty: The set of carried traffic intensity vectors is clearly nonempty.

Closed: In an asymptotic limit where the number of channels and the offered traffic are made arbitrarily large while keeping the ratio finite, and which we consider in the next section, for the model we have assumed (underlying model of offered traffic independent from cell to cell, etc.), the set of carried traffic intensity vectors is given by the following inequalities (Refer [2]):

$$x_i \leq \sum_{j=1}^{M} X_j a_{ij} \quad i = 1, 2, \ldots N \tag{1}$$

$$X_j \geq 0 \tag{2}$$

$$\sum_{j=1}^{M} X_j = 1 \tag{3}$$

$$x_i \leq p_i r \quad i = 1, 2, \ldots N \tag{4}$$

$$x_i \geq 0 \quad i = 1, 2, \ldots N \tag{5}$$

where $a_{ij} = 0$ or 1 depending on the system configuration and channel reuse. The a_{ij} can be found by modelling the system by a hypergraph [2]. This set is closed. For the finite channel case, we could not prove that the set of carried traffic intensity vectors, F, is closed. This remains an assumption. However even if the assumption turns out to be invalid our result is not seriously weakened. It can be shown that the closure of any convex, coordinate convex and bounded set is also convex, coordinate convex and bounded and obviously closed. Also clearly closure of a nonempty set is nonempty. Thus the closure of the set of carried traffic intensity vectors is nonempty, convex, coordinate convex, bounded (we are going to show that the set of carried traffic intensity vectors is convex, coordinate convex and bounded) and closed subset of R^N and hence has an MMF, $\tilde{x}^M$ (by Theorem 2). Even if $\tilde{x}^M$ does not belong to F (closedness is not necessary for the existence of MMF, $\tilde{x}^M$ may belong to F), we have carried traffic intensity vectors in F arbitrarily close to $\tilde{x}^M$. For practical purposes, it suffices to assume that the set of carried traffic intensity vectors is closed.

The set of carried traffic intensity vectors, F is bounded. This is because the carried traffic intensity is nonnegative in any cell and cannot exceed 1 since traffic carried in any cell can never exceed the total number of channels available in the system. Thus $0 \leq x_i \leq 1$ for each i and hence $0 \leq c_i \leq 1/p_i$ for each i, where $0 < p_i < 1$.

Theorem 3 *The set of achievable carried traffic intensity vectors for a cellular system, F is convex.*

We give an outline of the proof of this important result. Our assumptions about channel assignment algorithms, the call arrival process and call duration distribution allow the operation of the channel assignment algorithms to be modelled by a continuous time Markov chain (CTMC). Furthermore we assume the CTMC's to be positive recurrent and irreducible. Let there be two carried traffic intensity vectors $\tilde{x}^A$ and $\tilde{x}^B$ realized by channel assignment algorithms A and B respectively. Let $S_A = \{A_1, A_2, \ldots, \}$ (resp.

$S_B = \{B_1, B_2, \ldots, \})$ denote the state space of A (resp. B). Denote one null state (state in which the number of calls in progress in the system is 0) of A by T_A and one of B by T_B. A third channel assignment algorithm C whose state space is the union of S_A and S_B operates as follows. When in T_A it switches over to T_B at rate γ_{AB} and when in T_B it switches over to T_A at rate γ_{BA}. Otherwise C behaves in the same manner as A or B depending on whether it is in a state that belongs to S_A or S_B respectively (all other transition rates remain the same). It can be shown that the CTMC corresponding to C is positive recurrent and irreducible. $\tilde{x}^C = \alpha\tilde{x}^A + (1-\alpha)\tilde{x}^B$, where $\alpha = (1+\gamma\pi^A_{T_A}/\pi^B_{T_B})^{-1}$, $\gamma = \gamma_{AB}/\gamma_{BA}$, and $\{\pi^A\}$ and $\{\pi^B\}$ are the steady state probability distributions of algorithms A and B respectively. By adjusting γ we can get any $\alpha \in (0,1)$.

Theorem 4 *The set of achievable carried traffic intensity vectors for a cellular system, F is coordinate convex.*

A slight variation of the technique used to prove Theorem 3 in conjunction with Theorem 3 can be used to prove this theorem.

Thus the set of achievable carried traffic intensity vectors for a cellular system, is closed, bounded, convex, and coordinate convex. It is also a subset of R^N. Hence it has an MMF by theorem 2.

Theorem 5 *MMF is unique.*

Theorem 1 can be used to show that fairness is an antisymmetric relation. Uniqueness of MMF follows from this and the definition of MMF.

4 Properties and Computation of MMF in the Asymptotic Limit

In this section we shall consider an asymptotic limit where $A_i \to \infty$ and $n \to \infty$ but

$$\lim_{A_i \to \infty, n \to \infty} A_i/n = p_i r$$

is finite. Thus both the offered traffic and the number of channels are made arbitrarily large while keeping the ratio—the offered traffic intensity—finite. The properties of channel assignment algorithms in this asymptotic limit were studied in [2]. In this asymptotic limit, the MMF has many interesting properties. We shall study these in this section. Let $\tilde{x}^{M\infty}(r)$ denote the MMF carried traffic intensity vector at load r in this asymptotic limit. In this limit, the set of achievable carried traffic intensity vectors $F_\infty(r)$ at r is described by inequalities (1) to (5) for our model [2].

Theorem 6 *There exist finite nonnegative $\{\beta_i\}$ such that for each $i \in U$ and $r \geq 0$*

$$c_i^{M\infty}(r) = \min(r,\ \beta_i).$$

Using inequalities (1) to (5), the construction given after the statement of Theorem 2. and uniqueness of the MMF, it can be shown that $c_i^{M\infty}(r)$ is a nondecreasing function of r. The theorem follows from this nondecreasing property and inequalities (1) to (5).

Using inequalities (1) to (5), we can construct $\tilde{x}^{M\infty}(r)$ at any $r \geq 0$ following the construction given after the statement of Theorem 2.

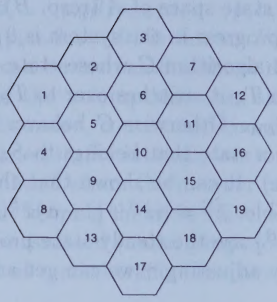


Figure 3: A 19-cell system

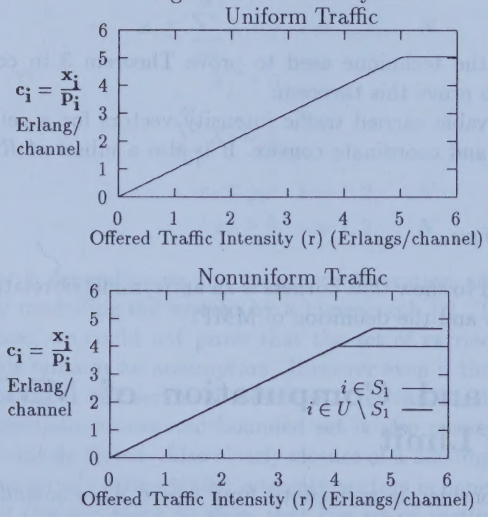


Figure 4: Maxmin fair c_i in a 19−cell system. $S_1 = \{5, 6, 9, 10, 11, 14, 15\}$ (central cell and its neighbors).

If $\tilde{x}^{M_\infty}(r)$ is found at some sufficiently large r, say r' for which $c_i^{M_\infty}(r') < r'$, for all $i \in U$, then $c_i^{M_\infty}(r') = \beta_i$, for all $i \in U$, from Theorem 6. Thus the β_i's can be determined from $\tilde{x}^{M_\infty}(r')$. Using these β_i's, $\tilde{x}^{M_\infty}(r)$ can be determined for all $r \geq 0$ from Theorem 6. Thus $\tilde{x}^{M_\infty}(r)$ need not be found separately at all values of r. (Using inequalities (1), (3) and (4), for any $r > 1/\min_{i \in U} p_i$, $c_i^V < r$, for all $i \in U$ and any $\tilde{x}^V \in F_\infty(r)$. Hence $c_i^{M_\infty}(r) < r$ for each $i \in U$.)

Theorem 7 *The carried traffic intensity vector corresponding to the fixed channel assignment algorithm which allocates $\lfloor np_i\beta_i \rfloor$ channels to cell i, for all $i \in U$, is the MMF for all $r \geq 0$, in the asymptotic limit.*

This result can be proved using Theorem 6.

Example 4.1 Consider the system of 19 cells (see Figure 3) with the following reuse constraint. A set of cells can use the same channel simultaneously if the interference in each cell of the set is $\leq 3/8$. The interference in any cell u is given by $\sum_{v \in S, v \neq u} d(u, v)^{-4}$,

where S is the set of cells using the same channel simultaneously and $d(u,v)$ is the distance between the centers of cells u and v. We assume the cell radius to be $1/\sqrt{3}$ or equivalently, the distance between the centers of adjacent cells to be unity. (The reuse constraints are the same as those in Example 1.2 of [2].)

1. We shall consider uniform traffic, i.e., $p_i = 1/19$ for each cell $i \in U$. The carried traffic intensity in the system under a maxmin fair channel assignment algorithm is given by

$$T_{M_\infty}(r) = \sum_{i=1}^{N} x_i^{M_\infty}(r) = \min(r,\ 247/49)$$

(see Figure 5 (Uniform Traffic)). $x_i^{M_\infty}(r) = \min(r/19,\ 13/49)$, $\forall i \in U$. Thus $c_i^{M_\infty}(r) = x_i^{M_\infty}(r)/p_i = \min(r,\ 247/49)$, $\forall i \in U$. In Figure 4 (Uniform Traffic) these c_i have been plotted. A fixed channel allocation algorithm allocating $\lfloor 13n/47 \rfloor$ channels to each cell achieves the MMF carried traffic intensity vector at all r, in the asymptotic limit.

2. Let us consider nonuniform traffic.

$$p_i = \begin{cases} 1/24, & i \in \{1,\ 2,\ 3,\ 4,\ 7,\ 8,\ 12,\ 13,\ 16,\ 17,\ 18,\ 19\} \\ 1/16, & i \in \{5, 6, 9, 11, 14, 15\},\ \text{and} \\ 1/8, & i = 10. \end{cases}$$

The maxmin fair carried traffic intensity in the system, $T_{M_\infty}(r) = \min(r,\ 2 + \frac{r}{2},\ \frac{13}{3})$ (refer Figure 5 (Nonuniform Traffic)). $x_i^{M_\infty}(r) = p_i \min(r,\ \beta_i)$, $\forall i \in U$.

$$\beta_i = \begin{cases} 14/3, & i \in \{1,\ 2,\ 3,\ 4,\ 7,\ 8,\ 12,\ 13,\ 16,\ 17,\ 18,\ 19\}, \\ 4, & \text{otherwise.} \end{cases}$$

In Figure 4 (Nonuniform Traffic), c_is have been plotted for these cells. The following fixed channel allocation algorithm gives the MMF at all r in the asymptotic limit. (n_i is the number of channels allocated to the ith cell.)

$$n_i = \begin{cases} \lfloor 7n/36 \rfloor, & i \in \{1,\ 2,\ 3,\ 4,\ 7,\ 8,\ 12,\ 13,\ 16,\ 17,\ 18,\ 19\}, \\ \lfloor n/4 \rfloor, & i \in \{5,\ 6,\ 9,\ 11,\ 14,\ 15\},\ \text{and} \\ \lfloor n/2 \rfloor, & i = 10. \end{cases}$$

5 Tradeoff between Fairness and Revenue Maximization

More often than not MMF is not the carried traffic intensity vector which yields the maximum revenue. At present we consider the rate per call per unit time to be uniform throughout the system and hence the total carried traffic intensity in the system gives the rate at which revenue is earned, except for a multiplicative constant which we consider as unity. We define the *marginal revenue* (see [7]) of a channel assignment algorithm A (or the corresponding carried traffic intensity vector $\tilde{x}^A(r)$) at load r and number of channels n, as

$$T_A(r,n) = \sum_{i=1}^{N} x_i^A(r).$$

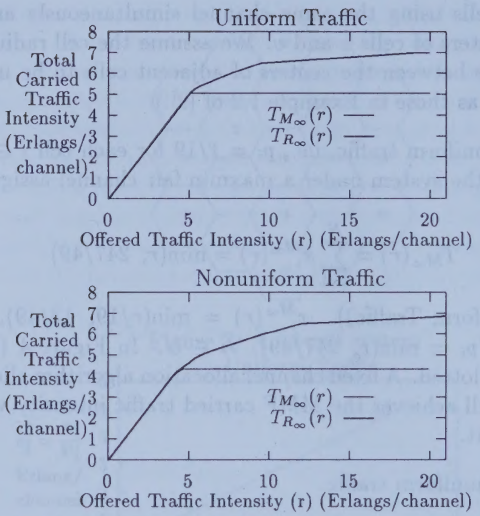


Figure 5: Loss in revenue brought about by maxmin fairness for a 19−cell system.

So the marginal revenue of a channel assignment algorithm is the rate at which revenue is earned per channel using that algorithm. Let $T_M(r,n)$ be the marginal revenue of the MMF. Let $T_R(r,n) = \sum_{i=1}^{N} x_i^R(r)$ be the maximum marginal revenue [7] and $R(r)$ the channel assignment algorithm ($\tilde{x}^R(r)$ the corresponding carried traffic intensity vector at load r) yielding this marginal revenue at load r. Let $T_{M_\infty}(r)$ and $T_{R_\infty}(r)$ be the corresponding asymptotic marginal revenues. Consider the following examples.

Example 5.1

1. Consider the system of Example 4.1(1). $T_{M_\infty}(r)$ and $T_{R_\infty}(r)$[‖] have been plotted vs r in Figure 5 (Uniform Traffic). For $r > 247/49$, $T_{M_\infty}(r) < T_{R_\infty}(r)$.

2. Consider the system of Example 4.1(2). For $r > 4.0$, $T_{M_\infty}(r) < T_{R_\infty}(r)$ (refer Figure 5 (Nonuniform Traffic)).

Clearly $T_M(r,n) \leq T_R(r,n)$ for all systems. Generally this inequality is strict. Whenever this inequality is strict, the channel assignment algorithm attaining $T_R(r,n)$ is unfair compared to that attaining $T_M(r,n)$. In other words, revenue is maximized at the expense of fairness. If the operator maximizes revenue at the cost of fairness, it may cause customer dissatisfaction (customers will be dissatisfied throughout the system as they will experience poor quality service when they move to certain cells) and possibly customer desertion to a competitor operator, which will result in reduced r and hence reduced revenue. Instead if the network operator had opted for a carried traffic intensity vector which fetches less revenue but is fairer, customer desertion may have been less and it is possible that the ultimate revenue earned would be greater. While we are still in the process of modelling the effects of customer desertion on fairness and revenue, it seems plausible that the ideal channel assignment algorithm in many situations would earn less

[‖]Computation of $T_{R_\infty}(r)$ has been discussed in [2].

marginal revenue than $T_R(r,n)$ (which we get by not caring about fairness) and more than $T_M(r,n)$ (which we get by being maxmin fair). It would also, in some sense (not necessarily that of section 2), be "fairer" than $R(r)$ but not "as fair" as the maxmin fair channel assignment algorithm $M(r)$ at load r. A possible approach is to fix the revenue at some value between $T_M(r,n)$ and $T_R(r,n)$, say m, and choose the channel assignment algorithm corresponding to the MMF among those carried traffic intensity vectors that *earn a marginal revenue of at least* m. We denote this carried traffic intensity vector by $\tilde{x}^M(m,r,n)$ ($\tilde{x}^{M_\infty}(m,r)$ in the asymptotic limit). Consider the following example.

Example 5.2 1. Consider the system of 19 cells with uniform traffic (refer Example 4.1(1)). At $r = 38/7$, $T_{R_\infty}(r) = 37/7 = 5.2856$ and $T_{M_\infty}(r) = 247/49 = 5.0408$. A carried traffic intensity vector $\tilde{x}^{R_\infty}(r)$ which maximizes the marginal revenue in the asymptotic limit (computed as per [2]) has a blocking probability of 50% in the central cell (cell 10) and 0% in all other cells. On the other hand $\tilde{x}^{M_\infty}(m,r)$ at $m = 5.2$ gives following blocking probabilities:

$$b_i = \begin{cases} 11.175\%, & i \in \{5,6,9,10,11,14,15\}, \\ 0.287\%, & i \in \{2,4,7,13,16,18\}, \\ 0\%, & \text{otherwise.} \end{cases}$$

Clearly $\tilde{x}^{M_\infty}(m,r)$ is much fairer than $\tilde{x}^{R_\infty}(r)$, while it fetches only slightly less revenue (5.2 as compared to 5.286) at $r = 38/7$. This suggests that a network operator may be better off in the long run sacrificing the additional revenue achieved by $\tilde{x}^{R_\infty}(r)$ and using $\tilde{x}^{M_\infty}(5.2,r)$ instead, at $r = 38/7$.

2. Consider the system of 19 cells with nonuniform traffic (refer Example 4.1(2)). At $r = 4.5$, $T_{R_\infty}(r) = 4.348$ and $T_{M_\infty}(r) = 4.25$. A carried traffic intensity vector $\tilde{x}^{R_\infty}(r)$ which maximizes the marginal revenue in the asymptotic limit (computed as per [2]) has a blocking probability of 26.98% in the central cell (cell 10) and 0% in all other cells. On the other hand $\tilde{x}^{M_\infty}(m,r)$ at $m = 4.3$ gives following blocking probabilities:

$$b_i = \begin{cases} 15.553\%, & i = 10 \\ 6.668\%, & i \in \{5,6,9,11,14,15\}, \\ 0\%, & \text{otherwise.} \end{cases}$$

Clearly $\tilde{x}^{M_\infty}(4.3,r)$ is much fairer than $\tilde{x}^{R_\infty}(r)$, while it fetches only slightly less revenue (4.3 as compared to 4.348) at $r = 4.5$. Again this suggests that a network operator may be better off in the long run sacrificing the additional revenue achieved by $\tilde{x}^{R_\infty}(r)$ and using $\tilde{x}^{M_\infty}(4.3,r)$ instead, at $r = 4.5$. MMF gives a blocking probability of 11.11% in cells $5,6,9,10,11,14,15$ and 0% blocking in other cells.

The value of m could possibly be the result of modelling customer desertion in an appropriate manner. We are currently working on this problem.

We have so far assumed that the set of carried traffic intensity vectors $F(m,r,\tilde{p},n)$ which fetch a marginal revenue of at least m at r has an MMF. This can be proved using the following theorem.

Theorem 8 *If D is any nonempty, convex, closed, bounded subset of R_+^N, D has an MMF.*

Note that the assumption of coordinate convexity is not required.

6 Conclusions and Future Work

We have developed the notions of relative and absolute fairness. We have obtained a simple necessary and sufficient condition for relative fairness which is useful in determining if a vector is fairer than another. We have shown that any subset of R^N satisfying certain properties has an MMF and the set of carried traffic intensity vectors satisfies those properties. We have shown that the MMF is unique. We have been able to specify a fixed channel assignment algorithm which yields the MMF at all loads, in the asymptotic limit. We are currently working on determining optimal channel assignment algorithms that would maximize revenue taking into account the effects of customer desertion.

References

[1] C. Berge, *Hypergraphs.* Amsterdam: North-Holland,1989.

[2] R. J. McEliece and K. N. Sivarajan, " Performance Limits for Channelized Cellular Telephone Systems," *IEEE Transactions on Information Theory*, vol. 40, no. 1, Jan. 1994.

[3] V. H. Macdonald, "The Cellular Concept," *The Bell System Technical Journal*, vol. 58, no. 1, January 1979.

[4] D. Bertsekas and R. Gallager, *Data Networks*, Englewood Cliffs, NJ:Prentice-Hall, 1987.

[5] E. Kreyszig, *Introductory Functional Analysis with Applications*, John Wiley and Sons, 1978.

[6] R. W. Wolff, *Stochastic Modeling and The Theory of Queues*, Englewood Cliffs, NJ:Prentice-Hall, 1989.

[7] R. J. McEliece and K. N. Sivarajan, " Maximizing Marginal Revenue in Generalized Blocking Service Networks," *Proc. 30th Annual Allerton Conference on Communications, Control and Computing*, vol. 18, 1992, pp. 455–464.

Statistical Multiplexing of Asynchronous Regulated Traffic*

Hui Jin
Robert J. McEliece
Zhong Yu

Department of Electrical Engineering
California Institute of Technology
Pasadena, California 91125, USA

Abstract. *In this semi-tutorial paper, we shall present an abstract model for studying some of the traffic congestion problems associated with ATM and similar multiaccess networks, in which the quality of service is measured by the average queue length. After noting that if the traffic streams are limited in total rate but otherwise uncontrolled, the worst-case performance is abysmal because of bursts of short duration, we restrict our attention to traffic that is controlled so as to satisfy a so-called (σ, ρ)-constraint. We then obtain fairly tight upper and lower bounds, in terms of σ and ρ, for the worst-case average queue length for a $D/G/1$ queue whose input satisfies a (σ, ρ) constraint. These bounds immediately imply reasonably good upper bounds on the average queue length for multiplexed (σ, ρ) traffic streams. Finally, we shall show that these bounds can be improved if we take into consideration the multiplexing, at least for a class of traffic models we call cyclically symmetric periodic (c.s.p.).*

1. Introduction and Summary.

In this paper, we consider the following simplified model of a multiaccess communications system. Time is slotted, and information comes in discrete quanta, called *cells*. The channel is capable of transmitting, noiselessly, exactly one cell per slot. There are K independent information sources who must share the channel. A single traffic stream (message source) is modelled as a random sequence of 0's and 1's, denoted $X(1), X(2), \ldots$. The event $X(n) = 1$ represents the generation of a message cell at time n, and $X(n) = 0$ means that no cell is generated at time n. The *rate* of such a source is defined as

$$\rho = \limsup_{n\to\infty} \frac{E(X_1) + \cdots + E(X_n)}{n}. \tag{1.1}$$

The rate ρ is measured in the (dimensionless) units of *cells per slot.*

It is the object of this paper to study the *statistical multiplexing* of two or more such sources. By this we mean the following. If there are K sources $X_1(n), \ldots, X_K(n)$, the outputs of the sources are presented simultaneously to the channel. Thus the number of cells arriving at the channel at time n is

$$X(n) = X_1(n) + \cdots + X_K(n). \tag{1.2}$$

We shall call the process $X(n)$ the *aggregate traffic.* If the ith source has rate ρ_i, then by (1.1), the combined source rate is $\rho = \rho_1 + \cdots + \rho_K$ cells per slot.

Cells which cannot be transmitted immediately are queued (first-in, first out) for transmission in the first available slot. In this way the system is, in effect, a $D/G/1$ queue, with constant interarrival time equal to one, and service times given by the sequence of nonnegative integers $X(n)$ in (1.2).

---

* This work was supported by a grant from Pacific Bell.

The *quality of service* of the system is governed by the size of the channel backlog, i.e., the number of cells waiting in the queue. Denoting the number of waiting cells at time n by Q_n, we have the fundamental recursive formula

$$\begin{aligned} Q_0 &= 0 \\ Q_n &= \max(Q_{n-1} + X(n) - 1, 0). \end{aligned} \tag{1.3}$$

In this paper, we will measure the quality of service by the *average* queue length $\overline{Q}$, which is defined as

$$\overline{Q} = \limsup_{n \to \infty} \frac{E(Q_1) + \cdots + E(Q_n)}{n}. \tag{1.4}$$

All else being equal, we will prefer a system with a small value of $\overline{Q}$ to one with a large value of $\overline{Q}$.

Here is an outline of the rest of the paper. In Section 2, we shall identify the "worst-case" (largest $\overline{Q}$) in the special (and unrealistic) case of independent Bernoulli traffic processes. The worst case, not surprisingly, turns out to be when the aggregate traffic is Poisson, which is achievable in the limit of a large number of independent identically distributed, low-rate traffic streams, for which we have $\overline{Q} = \rho^2/2(1-\rho)$. We shall also show with a simple example that for "synchronized" traffic, for any $\rho > 0$, $\overline{Q}$ can be arbitrarily large. This is because of the possibility of long bursts in the aggregate traffic, even if the individual sources are burst-free. To prevent this pathological behavior, we assume in the remainder of the paper that each traffic source is "controlled." The particular control scheme is not important, so long as each of the traffic sources satisfies the burst-limiting "(σ, ρ)" constraint first introduced by Cruz [2]. In Section 3, we then obtain fairly tight upper and lower bounds, in terms of σ and ρ, for the worst-case $\overline{Q}$ when the aggregate traffic that satisfies a (σ, ρ) constraint. Finally, in Section 4 we further restrict our traffic to the class of "cyclically symmetric periodic" processes, a class which models the property, observed in practice, that the individual traffic streams cannot be "synchronized." We then obtain some results about $\overline{Q}$ for c.s.p. traffic, which shows that c.s.p. traffic is better behaved, especially at low rates, than unrestricted (σ, ρ) traffic.

2. Some Simple Observations.

The simplest case of statistical multiplexing is when the K sources are independent, Bernoulli sources. Here the ith traffic stream, $X_i(1), X_i(2), \ldots$ is a sequence of independent, identically distributed binary random variables with

$$\begin{aligned} \Pr\{X_i(n) = 1\} &= \rho_i \\ \Pr\{X_i(n) = 0\} &= 1 - \rho_i. \end{aligned} \tag{2.1}$$

The combined traffic is also i.i.d. with rate $\rho = \rho_1 + \cdots + \rho_K$. An elementary calculation shows that in this case, the average queue size is

$$\overline{Q}(\rho_1, \ldots, \rho_K) = \frac{\sum_{i<j} \rho_i \rho_j}{1 - \rho}. \tag{2.2}$$

The expression in (2.2) is maximized when $\rho_1 = \cdots = \rho_K = \rho/K$, in which case the combined traffic is binomially distributed, with average queue length

$$\overline{Q}_K = \frac{\rho^2}{2(1-\rho)} (1 - 1/K). \tag{2.3}$$

In the limit as $K \to \infty$, i.e., infinitely many sources each with infinitesimal rate, the combined traffic becomes Poisson, and the corresponding mean queue length is

$$\overline{Q} = \frac{\rho^2}{2(1-\rho)}. \tag{2.4}$$

These elementary considerations are summarized in the following theorem. .

2.1 Theorem. *For K independent Bernoulli traffic streams, with combined rate ρ,*

$$\overline{Q} < \frac{\rho^2}{2(1-\rho)}. \tag{2.5}$$

Furthermore, equality is asymptotically achieved in (2.5) in the limit as $K \to \infty$, if the traffic streams are identically distributed, in which case the combined traffic is asymptotically Poisson.

What Theorem 2.1 tells us is that (at least if the quality of service is measured by $\overline{Q}$), Poisson traffic is the worst possible traffic with *independent cell arrivals.* In a sense, the goal of this paper is to discover the worst possible traffic if the i.i.d. assumption is dropped. Unfortunately, however, if the traffic streams are not independent, the aggregate traffic can be arbitrarily bursty, and things can be much worse, as the following example demonstrates.

2.2 Example. Consider K identical deterministic sources, each of period L:

$$X_i(n) = \begin{cases} 1 & \text{if } n \text{ is a multiple of } L \\ 0 & \text{if not.} \end{cases}$$

Then the combined traffic is given by

$$X(n) = \begin{cases} K & \text{if } n \text{ is a multiple of } L \\ 0 & \text{if not.} \end{cases}$$

The rate of the traffic is then $\rho = K/L$. Using the definition (1.3), we find that the queue length is periodic with pattern

$$K-1, K-2, \ldots, 1, 0, \cdots, 0,$$

so that the average queue length is (see (1.4))

$$\begin{aligned} \overline{Q} &= \frac{K(K-1)}{2L} \\ &= \rho \cdot \frac{K-1}{2} \approx \frac{\rho^2}{2} \cdot L. \end{aligned} \tag{2.6}$$

■

The result of Example 2.2 should be compared with that of Theorem 2.1. Theorem 2.1 says that for Bernoulli traffic of rate ρ, $\overline{Q}$ is bounded by $\rho^2/2(1-\rho)$. This behavior is reasonable at low rates, but unsatisfactory as the channel capacity is reached, i.e., as $\rho \to 1$. On the other hand, Example 2.2 shows that if the aggregate traffic is bursty, then for any value of $\rho > 0$, $\overline{Q}$ can be arbitrarily large.* In both cases, clearly in Example 2.2, but less so in Theorem 2.1, the large values of $\overline{Q}$ are caused by occurrence of long, high-rate, bursts of aggregate traffic. Thus if we could somehow control the burstiness of the traffic, we could ensure a relatively small value of $\overline{Q}$. In the rest of the paper, therefore, we will restrict our attention to a class of non-i.i.d. traffic sources for which the "burstiness" is controlled.

3. (σ, ρ) Regulated Traffic.

We have seen in Section 2 that if the traffic is sufficiently bursty, the average queue length can be arbitrarily large. To prevent this, some method is required to "regulate" the traffic. There have been many suggestions as to how this may best be done. One of the most promising and general methods is to require that each traffic stream be a "(σ, ρ) sequence," a concept introduced by Cruz [2]. (One way to enforce this is by using a "leaky bucket" controller [9][13][14][15].) In this section we will define the class of (σ, ρ) sequences, and develop bounds on the average queue length for a D/G/1 queue with (σ, ρ) service times.

3.1 Definition. *A sequence $X(1), X(2), \ldots$ of nonnegative integers is said to be a (σ, ρ) sequence, if, for all integers $n \geq 1$ and $L \geq 1$,*

$$X(n+1) + \cdots + X(n+L) \leq \sigma + \rho L. \tag{3.1}$$

Similarly, a random process $(X(n))$ is said to be a (σ, ρ) process if (3.1) is almost surely true of each subsequence of the form $X(n+1), \ldots, X(n+L)$.

Note that if the ith traffic stream is a (σ_i, ρ_i) sequence, then the aggregate traffic is a (σ, ρ) sequence, with

$$\sigma = \sum_i \sigma_i, \qquad \rho = \sum_i \rho_i,$$

Thus any results we can obtain about the average queue length for a $D/G/1$ queue whose input process satisfies a (σ, ρ) constraint will shed light on the behavior of statistical multiplexed tfaffic where each of the sources satisfies a (σ_i, ρ_i) constraint.

A simple but fundamental result about (σ, ρ) sequences is the following.

3.2 Theorem. *If the service times for a $D/G/1$ queue with interarrival time equal to 1 form a (σ, ρ) sequence with $\rho \leq 1$, then $Q_n \leq \sigma$ for all $n \geq 1$.*

Proof: Using the fundamental recursion (1.3), we can prove by induction that

$$Q_n = \max_{1 \leq k \leq n} \left(\sum_{j=k}^{n} X(j) - n + k - 1 \right). \tag{3.2}$$

---

* The reader may object that in this example the users are synchronized and conspiring against the server. In Section 4 we will see that the same kind of thing can be demonstrated for asynchronous traffic. Our result (Theorem 4.10) gives $\overline{Q} \approx \rho^3 L/8$.

Thus we have

$$\begin{aligned} Q_n &= \max(\sum_{j=k}^{n} X(j) - n + k - 1) \qquad \text{(by (3.2))} \\ &\le \max(\sigma + (\rho - 1)(n - k + 1)) \qquad \text{(by the } (\sigma, \rho) \text{ property)} \\ &\le \sigma. \qquad \text{(since } \rho \le 1) \quad \blacksquare \end{aligned}$$

What Theorem 3.2 says is that if the service time process for a D/G/1 queue is a (σ, ρ) sequence, and if the buffer has capacity σ or greater, buffer overflow cannot occur. While this is reassuring if σ is small, for large values of σ (which corresponds to very bursy traffic) it is necessary to know something about the backlog *distribution* which can be produced by (σ, ρ) traffic. In this direction, we have the following sequence of theorems and corollaries, whose proofs will appear in a longer paper [7].

3.3 Theorem. *Denote by $\overline{Q}(\sigma, \rho)$ the maximum possible average queue occupancy for a $D/G/1$ queue with interarrival time equal to 1 whose service times form an (σ, ρ) process with $0 \le \rho \le 1$. Then*

$$\overline{Q}(\sigma, \rho) \le \rho \left(2\sigma + 1/2 - \rho\sigma - \sqrt{(2 - \rho)(1 - \rho)(\sigma^2 + \sigma)}\right).$$

3.4 Corollary. *If $0 \le \rho \le 1$, is fixed, then*

$$\lim_{\sigma \to \infty} \frac{\overline{Q}(\sigma, \rho)}{\sigma} \le \rho \left(2 - \rho - \sqrt{(2 - \rho)(1 - \rho)}\right).$$

3.5 Theorem. *With the same setup as Theorem 3.3, if $0 \le \rho < 1/2$, then*

$$\overline{Q}(\sigma, \rho) \le 3\sigma\rho \left(3 - 2\rho - \sqrt{2(3 - 2\rho)(1 - \rho)}\right) + \rho^2/2.$$

3.6 Corollary. *If $0 \le \rho < 1/2$ is fixed, then*

$$\lim_{\sigma \to \infty} \frac{\overline{Q}(\sigma, \rho)}{\sigma} \le \rho \left(3 - 2\rho\sqrt{2(3 - 2\rho)(1 - \rho)}\right).$$

3.7 Theorem. *If $0 \le \rho < 1/n$ $(n \ge 3)$ is fixed, then*

$$\lim_{\sigma \to \infty} \frac{\overline{Q}(\sigma, \rho)}{\sigma} \le \rho \left((n + 1 - n\rho) - \sqrt{n(n + 1 - n\rho)(1 - \rho)}\right).$$

The lower envelope of the sequence of asymptotic upper bounds in Corollaries 3.4, 3.6, and Theorem 3.7 is shown in Figure 1.

Next, we present a sequence of (asymptotic) lower bounds on $\overline{Q}(\sigma, \rho)$. These bounds, which are derived in detail in [7], are obtained by explicitly constructing (σ, ρ) sequences which have the average queue lengths as indicated:

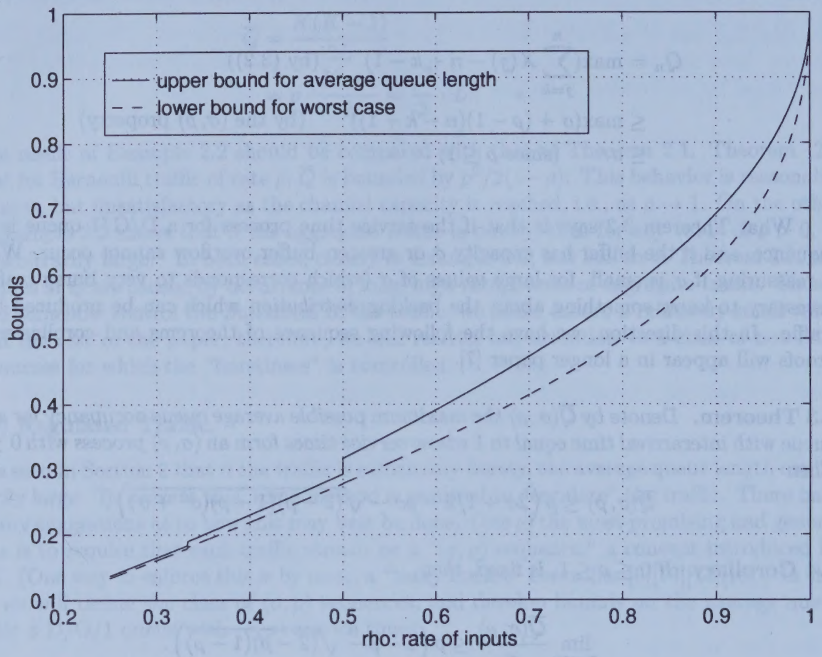


Figure 1. The asymptotic upper and lower bounds obtained on the worst-case $\overline{Q}/\sigma$ for fixed ρ and large σ.

3.8 Theorem. *If* $1/2 \le \rho < 1$, *then*

$$\lim_{\sigma\to\infty} \frac{\overline{Q}(\sigma,\rho)}{\sigma} \ge \rho\left(3-2\rho-\sqrt{(2-\rho)(1-\rho)}-\frac{3-2\rho}{2}\sqrt{\frac{1-\rho}{2-\rho}}\right).$$

3.9 Theorem. *If* $1/3 \le \rho < 1/2$, *then*

$$\lim_{\sigma\to\infty} \frac{\overline{Q}(\sigma,\rho)}{\sigma} \ge \rho\left(4-3\rho-\frac{3}{2}\sqrt{\frac{(2-\rho)(1-\rho)}{2}}-\frac{4-3\rho}{2}\sqrt{\frac{2(1-\rho)}{3-2\rho}}\right)$$

$$\vdots$$

3.10 Theorem. *If* $1/n \le \rho < 1/(n-1)$, *then*

$$\lim_{\sigma\to\infty} \frac{\overline{Q}(\sigma,\rho)}{\sigma} \ge \rho\left(n+1-n\rho-\frac{n+1-n\rho}{2}\sqrt{\frac{(n-1)(1-\rho)}{n-(n-1)\rho}}-\frac{n}{2}\sqrt{\frac{(1-\rho)(n-(n-1)\rho)}{n-1}}\right)$$

The upper envelope of these lower bounds is also shown in Figure 1. It is seen that the bounds are very close together, so in a sense we have "solved" the problem of finding the worst case (σ,ρ) traffic.

The results in this section immediately imply upper bounds on $\overline{Q}$ for statistically multiplexed traffic for which the aggregate traffic satisfies a (σ, ρ) constraint, for example when the ith stream satisfies a (σ_i, ρ_i) constraint, for $i = 1, \ldots, K$, with $\sigma_1 + \cdots + s_K = \sigma$ and $\rho_1 + \cdots + \rho_K = \rho$. For example, by Theorem 3.2, the buffer occupancy is bounded by σ, and so buffer overflow can be prevented with certainty if the buffer has capacity $\geq \sigma$. If however σ is large, then for example by Corollary 3.4, the combined traffic will satisfy

$$\overline{Q} \leq \sigma\rho\left(2 - \rho - \sqrt{(2-\rho)(1-\rho)}\right). \tag{3.3}$$

However, it seems unlikely that the worst-case (σ, ρ) traffic can be realized as the aggregate traffic formed from a sum of independent (σ_i, ρ_i) streams, so that the bound in (3.3) may be unduly pessimistic. Unfortunately, we have not been able to obtain useful analytical results about $\overline{Q}$ for a general traffic stream that is the sum of independent (σ_i, ρ_i) streams. However, in the next section we will introduce a class of "asynchronous" (σ_i, ρ_i) traffic sources which we believe to be realistic, and for which we can rigorously improve the bound in (3.3).

4. Cyclically Symmetric Periodic Traffic.

In this section, we introduce a class of models for "regulated" traffic which are (σ, ρ) processes, have random phase, and exhibit relatively severe burstiness. The random phase feature is a mathematical way of ensuring that the traffic sources cannot be synchronized with each other, thus forbidding the pathological behaviour we saw in Example 2.2. (This class of traffic models is the same as that introduced in [4].)

4.1 Definition. *Let* $\mathbf{X}$ *be a random binary vector of length* L*, with the following symmetry condition.*

If $\mathbf{x}$ *and* $\mathbf{y}$ *are cyclic shifts of each other, then* $\Pr\{\mathbf{X} = \mathbf{x}\} = \Pr\{\mathbf{X} = \mathbf{y}\}$.

Then if $\mathbf{X}^*$ *represents the vector obtained from* $\mathbf{X}$ *by continually repeating it, i.e.,* $\mathbf{X}^* = \mathbf{X} * \mathbf{X} * \mathbf{X} * \cdots$*, then* $\mathbf{X}^*$ *is called a cyclically symmetric periodic (c.s.p.) random process, of period* L*. If the sum of the components of* $\mathbf{X}$ *is* w*, the rate of* $\mathbf{X}^*$ *is defined to be* $\rho = w/L$.

4.2 Example. Let $L = 5$. Then the random sequence which is equally likely to be any of the following sequences is a cyclically symmetric random process of period 5 and rate $\rho = 2/5$.

$$\begin{aligned} X^* &= (1100011000\cdots) \text{ or any of its shifts with probability } 2/15. \\ &= (1010010100\cdots) \text{ or any of its shifts with probability } 1/15. \\ &= \text{anything else with } p = 0. \end{aligned}$$

The following simple lemma, whose proof we omit, shows that any c.s.p. process satisfies a (σ, ρ) constraint.

4.3 Lemma. *If* $(X(n))$ *is a cyclically symmetric random process of rate* ρ *and period* L*, then* $(X(n))$ *is a* (σ, ρ) *process, with* $\sigma = \rho L$.

Combining Lemma 4.3 with Theorem 3.2, we see that if the traffic sources are all c.s.p. processes with rates $\rho_1, \ldots, \rho_K$ and common period L, then $Q_n \leq \rho L$ for all $n \geq 1$, so that trivially

$$\overline{Q} \leq \rho L = \sigma. \tag{4.1}$$

This bound can be improved, as follows:

4.4 Theorem. *If the traffic consists of K independent c.s.p. sources with common period L, then*

$$\overline{Q} \le \rho^2 L/2 = \sigma\rho/2.$$

Note that for values of ρ near 1, the bounds in (4.1) and Theorem 4.4 differ only by a factor of 2, while for small values of ρ, Theorem 4.4 is much stronger than (4.1). For *two* independent c.s.p. streams, the bound in Theorem 4.4 can still further:

4.5 Theorem. *For $K = 2$ independent c.s.p. sources of period L,*

$$\overline{Q} \le \frac{4\rho^3 - \rho^4}{16} L < \frac{\sigma\rho^2}{4}.$$

We have not suceeded in generalizing Theorem 4.6 for larger values of K. However, detailed calculations which we cannot yet rigorously justify lead us to the following conjecture:

4.7 Conjecture. *If the traffic consists of K independent c.s.p. sources with common period L, then*

$$\overline{Q} \le \frac{K-1}{2K}\rho^3 L + o(\rho^3 L) \approx \frac{K-1}{K}\sigma\rho^2,$$

for fixed K, as $\rho \to 0$.

We do not know if there are c.s.p. sources whose average queue length is as large as allowed by Theorem 4.6 or Conjecture 4.7, but we can can come reasonably close with the class of *greedy on-off sources*, which have been conjectured to be worst-case traffic sources in other contexts [15], [5].

4.8 Definition. *A "Greedy-On-Off" source with period L and rate ρ, is a c.s.p. process for which each of the L solid cyclic bursts of length $\lfloor \rho L \rfloor$ is assigned probability $1/L$, and all other sequences are assigned probability zero.*

4.9 Example. The following is a GOO source of period 5 and rate 2/5:

$$\begin{aligned} \mathbf{X}^* &= 1100011000\cdots \text{with probability } 1/5 \\ &= 0110001100\cdots \text{with probability } 1/5 \\ &= 0011000110\cdots \text{with probability } 1/5 \\ &= 0001100011\cdots \text{with probability } 1/5 \\ &= 1000110001\cdots \text{with probability } 1/5 \quad \blacksquare \end{aligned}$$

4.10 Theorem. *For $K = 2$, if the two traffic streams are independent GOO sources with period L and rate $\rho/2$ each, then*

$$\overline{Q} = \frac{\rho^3}{8} L = \frac{\sigma\rho^2}{8}. \tag{4.2}$$

We have been unable to find a closed-form expression for $\overline{Q}$ for $K \ge 3$ GOO sources. However, computer simulation suggests that for $K = 3$, the GOO source has $\overline{Q} \sim \alpha_3\rho^3 L$, with $\alpha_3 > 1/8$, and for $K = 4$, the GOO source has $\overline{Q} \sim \alpha_4\rho^3 L$, with $\alpha_4 < 1/8$. This leads to the odd conjecture that the worst-case GOO traffic occurs for $K = 3$!

One immediate implication Theorem 4.10 is that the upper bound in Theorem 4.6 is at least of the right order, viz. $O(\rho^3 L)$. Another implication is that for any $\rho > 0$, there exist

asynchronous traffic sources with aggregate rate ρ and arbitrarily large $\overline{Q}$. In a sense, this completes the picture begun in Example 2.2.

Finally, we note that in Theorem 2.1, we saw that for independent Bernoulli traffic, the worst case is a large number of identically distributed users. It is natural to conjecture that this is also true for c.s.p. traffic, i.e., that the globally worst case for c.s.p. traffic of aggregate rate ρ and common period L is a set of $K = \lfloor \rho L \rfloor$ "unit" GOO sources, i.e., GOO sources with only one cell per period. Surprisingly, however, this is not the case. We have been able to show [10] that for the set of unit GOO sources of common period L, that the average queue length satisfies

$$\overline{Q} = O(\sqrt{L}),$$

independent of ρ and K. Thus we are presently at a loss to guess the absolute worst case c.s.p. traffic. However, if we combine Conjecture 4.7 with Theorem 4.10, we see that the worst-case traffic satisfies

$$\frac{\rho^3}{8} L \leq \overline{Q} \stackrel{?}{\leq} \frac{\rho^3}{2} L,$$

which gives a reasonably good "order of magnitude" estimate.

5. Summary and Conclusions.

In this paper, we have studied the behavior of the average queue length when several "on-off" traffic sources of aggregate rate $\rho < 1$ are multiplexed together. We concluded that

- If the traffic sources are independent Bernoulli, then

$$\overline{Q}(\rho) \approx \frac{\rho^2}{2(1-\rho)}, \tag{5.1}$$

and is achieved by a very large number of very low-rate sources.

- If the individual traffic streams can be synchronized, then

$$\overline{Q}(\rho) = \infty. \qquad \text{for all } \rho < 0. \tag{5.2}$$

- Motivated by these results, we then restricted the traffic to satisfy a (σ, ρ)-constraint, for some $\sigma > 0$, which limits the "burstiness" of the aggregate traffic. For (σ, ρ)-traffic, we found by elementary considerations that

$$\overline{Q}(\sigma, \rho) \leq \sigma, \tag{5.3}$$

and by less elmentary considerations that

$$\overline{Q}(\sigma, \rho) \stackrel{\sim}{\leq} \sigma\rho\left(2 - \rho - \sqrt{(2-p)(1-p)}\right). \tag{5.4}$$

- Finally, by considering two independent "greedy on-off" sources, we showed that

$$\overline{Q}(\sigma, \rho) \geq \frac{\sigma\rho^2}{8}. \tag{5.5}$$

We note that for values of ρ near 1, the upper boound in (5.4) and the lower bound in (5.5) differe only by a factor of 8.

We conclude with a list of open problems. A solution to any one of them would be more than welcome.

- Generalize Theorem 4.10 to larger values of K. As a step in this direction, we have obtained a closed-form, but unwieldy, formula for $\overline{Q}$, which will appear in [10].

- Prove (or disprove) Conjecture 4.7.

- Generalize Theorem 4.6 to arbitrary $K \geq 2$.

- Prove of disprove the vague conjecture that independent "greedy on-off" sources are the worst case (σ, ρ) sources.

References.

1. E. Berlekamp and R. McEliece, "Average-case optimized buffered decoders," in *The Impact of Processing Techniques on Communications*, J. K. Skwirzynski, ed. (Vol. 91E in the NATO ASI Series), Dordrecht: Martinus Nijhoff (1985), pp. 145–158.

2. R. L. Cruz, "A Calculus for network delay, part I: network elements in isolation", *IEEE Trans. Inform. Theory*, Vol. 37, No. 1 (January 1991), pp. 114-131,

3. B. T. Doshi, "Deterministic rule based traffic descriptors for broadband ISDN: worst case behaviour and connection acceptance control," *Proc. 1994 International Teletraffic Congress.* Antibes, France, June 1994: vol. 1a, pp. 591–600.

4. A. E. Eckberg, Jr., "The single server queue with periodic arrival process and deterministic service times", IEEE Trans. Commun., Vol. COM-27, No. 3 (March 1979).

5. B. Ermili, J. Murphy, and J. Murphy, "Om worst case traffic in ATM networks," *Proc. 12th IEE Teletraffic Symp.*, London, March 1995.

6. D. Gross and C. M. Harris, *Fundamentals of Queueing Theory.* John Wiley & Sons, 1985.

7. H. Jin and R. J. McEliece, "Worst-case buffer occupancy for (σ, ρ)-regulated traffic," *in preparation.*

8. L. Kleinrock, *Queuing Systems, Volume 1.* New York: John Wiley & Sons, 1975.

9. T. Konstantopoulos and V. Anantharam, "Optimal flow control schemes that regulate the burstiness of traffic", *IEEE/ACM Trans. Networking*, Vol. 3, No. 4 (August 1995).

10. R. McEliece and Z, Yu, "Ballot theorems and queue occupancy for statistical multiplexers," *in preparation.*

11. R. O. Onvural, *Asynchronous Transfer Mode Networks: Performance Issues.* Artech House, 1994.

12. M. Sidi, W. Z. Liu, I. Cidon, and I. Gopal, "Congestion control through input rate regulation", *Proc. IEEE Globecom '89*, pp. 1764-1768.

13. K. Sohraby and M. Sidi, "On the performance of Bursty and Modulated sources subject to leaky bucket rate-based access control schemes", IEEE Trans. Commun., Vol. 42, No. 2/3/4 (February/March/April 1994), pp.477-487.

14. J. Turner, "New directions in communications (or which way to the information age?)", *IEEE Commun. Mag.*, Vol. 24, No. 10 (Oct. 1986), pp. 8-15.

15. T. Worster, "Modelling deterministic queues: the leaky bucket as an arrival process," *Proc. 1994 International Teletraffic Congress.* Antibes, France, June 1994: vol. 1a, pp. 581–590.

16. N. Yamanaka, Y. Sato, and K. Sato, "Performance limitation of leaky bucket algorithm for usgae parameter control and bandwidth allocation methods," *IEICE Trans. Comm.*, vol. E75-B, no. 2 (Feb. 1992).

OPTIMAL DISTRIBUTED ROUTING IN NETWORKS WITH RANDOMLY-VARYING CONNECTIVITY

Vahid Tabatabaee, Leandros Tassiulas
Department of Electrical Engineering
and Institute for Systems Research
University of Maryland
College Park, MD 20742

Extended Summary

In networks with mobile radio nodes, the connectivity varies rapidly with time. This is because of channel impairment and node mobility among other factors. As a consequence routing becomes a challenging task. Flooding is one of the approaches considered extensively for routing in these networks. While it is very effective for reliable and timely information delivery, it becomes fairly inefficient when the traffic load is substantial because of the high volume of redundant traffic that is generated. While certain types of information with high reliability and delivery time requirements may need be forwarded using flooding, this method is unsuitable for the large portion of the traffic that needs to be supported. Hence efficient single path routing algorithms need to be considered. In this paper, we summarize our recent results on distributed optimal routing in a network model with randomly changing topology. The scope in this paper is presentation of our approach and the main results without elaboration on the proofs which are parts of a full version of the paper.

The network is represented by a directed graph $G(I, L)$, where I is the set of nodes and L is the set of the links. The connectivity of the link from i to j is represented by the binary variable $C_{ij}(t)$. If $C_{ij}(t) = 1$, then a packet in node i may be forwarded to node j in one slot. We assume that the sequence of connectivity matrices $C(t)$, $t = 1, 2, \ldots$ is an i.i.d. process. The connectivity probability of link ij is

$$P(C_{ij}(t) = 1) = p_{ij}.$$

Node j is neighbor of node i if $p_{ij} > 0$. We also assume that $p_{ii} = 1$, this means that every node is always connected to itself.

A routing rule for node i is a function $R_i(t)$ whose output is the successor of node i at time t. A routing rule for node i is feasible if

$$R_i(t) = j \textit{ only when } C_{ij}(t) = 1$$

Assume that a packet starts from some arbitrary node of the network and needs to be forwarded to a prespecified destination node. A routing policy is a set of routing rules for all nodes at all times, $(R_i(t) : \quad i = 1, \ldots, N, \quad t = 1, 2, \ldots)$. Our goal is to find the optimal policy under which the expected distance from every node to a specified destination node d is minimized. We consider the class of policies for which at any time instant the successor of a node is selected based on the connectivity matrix $C(t)$ at time t. For these policies the movement of the packet is

†This research was supported in part by NSF under grant NCR-9406415, a CAREER award NCR-9502614, by the AFOSR under grant 95-1-0061 and ATIRP Consortium sponsored by the U.S. Army Research Laboratory under the Federated Laboratory Program, Cooperative Agreement DAAL01-96-2-0002.

a Markov chain on the nodes of the graph with the transition probabilities $q_{ij} = Pr(R_i = j)$. For all routing rules of interest this Markov chain has exactly one absorbing state, the destination node, while all other states are transient. The expected delivery time, D_i^R of a packet from node i, is the expected hitting time of d starting from node i of the above Markov chain. Hence the following equation is satisfied

$$D_i^R = 1 + \Sigma_{j=1}^N P(R_i = j) D_j^R \quad \forall i. \tag{1}$$

Consider two different routing rules R_1 and R_2. We say that R_1 is dominated by R_2 if and only if,

$$D_i^{R_2} \leq D_i^{R_1} \quad i = 1, \ldots, N \quad \textit{and} \quad D_l^{R_2} < D_l^{R_1} \quad \textit{for some } l$$

Consider the following condition on a routing rule R:

$$R_i(C) = \arg \min_{j:Cij=1} D_j^R \quad \textit{if } Pr[C(t) = C] > 0 \qquad (\mathcal{A})$$

Proposition 1 *If a routing rule R does not satisfy condition $(\mathcal{A})$, there is another routing rule R' that dominates R.*

Consider the mapping $\underline{M}(.)$ on R_+^N defined as follows:
Given a vector $\underline{X} \in R_+^N$ let $(i_1, \ldots, i_{k_i})$ be the ordered index set of those elements of $\underline{X}$ that are less than X_i. Moreover they are ordered in the set in increasing order of their element i.e.,

$$X_{i_1} \leq X_{i_2} \leq \ldots \leq X_{i_{k_i}} < X_i$$

The ith element of $\underline{M}(\underline{X})$ is

$$\begin{aligned} M_i(\underline{X}) &= 1 + P[C_{ii_1} = 1]\ X_{i_1} + P[C_{ii_1} = 0, C_{ii_2} = 1]\ X_{i_2} + \ldots \\ &+ P[C_{ii_1} = 0, \ldots, C_{ii_{k_i-1}} = 0, C_{ii_{k_i}} = 1]\ X_{i_{k_i}} \\ &+ P[C_{ii_1} = 0, \ldots, C_{ii_{k_i-1}} = 0, C_{ii_{k_i}} = 0]\ X_i \end{aligned} \tag{2}$$

where $C = (C_{ij} : i, j = 1, \ldots, N)$ is a random vector that follows the distribution of the connectivity matrix.

It can be verified that if a routing rule satisfies condition $(\mathcal{A})$ then the associated delivery time vector $\underline{D}^R$ satisfies the equation,

$$\underline{D}^R = \underline{M}(\underline{D}^R) \tag{3}$$

Also conversely, if a delivery time vector satisfies (3), then the associated routing rule R which has as delivery time vector $\underline{D}^R$ can be readily obtained from $(\mathcal{A})$.

With the help of proposition 1 the following can be shown.

Proposition 2 *The mapping M(.) has a unique fixed point satisfying equation(3). This is the vector of minimum delivery times and the associated routing policy is the optimal policy.*

Let $\underline{X}^n$, $n = 0, 1, 2, \ldots$ be a sequence of vectors derived by the iterative application of $\underline{M}(.)$ to a vector $\underline{X}^0$. The following holds.

Proposition 3 *The sequence $\underline{X}^n$, $n = 0, 1, 2, \ldots$ converges to the optimal delivery time vector from any initial positive vector $\underline{X}^0$.*

Note that the iterative application of $\underline{M}(.)$ accounts only to a simple operation performed by each node i, which involves only the elements X_j^n of the neighbors j of node i. Therefore, proposition 3 suggests a distributed policy for the computation of the minimum delivery time vector and the optimal policy. Using some results from [1] we can prove, that the convergence

to the minimum delivery time vector holds even when each node updates its own component of the vector asynchronously from the others. Hence the algorithm is robust.

The problem of routing in networks with changing topology has been studied in the past. In [2] and [3] algorithms for maintaining loop-free routing paths under changes on the topology were presented. In [4] and [5] results on the optimal routing problems in networks with stochastic variabilities were presented.

The objective of the work that is briefly reported in this paper, is to develop a methodology for the design of implementable routing algorithms which adapt to connectivity changes and have near optimal performance.

References

[1] D.P. Bertsekas and J.N. Tsitsiklis. *Parallel and distributed computation: Numerical methods.* PrenticeHall, 1989.

[2] M.S. Corson and A. Ephremides. A distibuted algorithm for mobile wireless networks. *Wireless Networks*, 1, 1995.

[3] E. Gafni and D. Bertsekas. Distributed algorithms for generating loop-free routes in networks with frequently changing topology. *IEEE Trans. Commun.*, Jan. 1981.

[4] R.G. Ogier and V. Rutenburg. Minimum-expected-delay alternate routing. *Infocom*, 1992.

[5] A. Orda, R. Rom, and M. Sidi. Minimum delay routing in stochastic networks. *IEEE/ACM Trans. Networking*, 1(2), 1993.

Probability Propagation and Iterative Decoding

Brendan J. Frey and **Frank R. Kschischang**
Department of Electrical and Computer Engineering
University of Toronto
brendan@comm.utoronto.ca and frank@comm.utoronto.ca

Abstract

In this paper, we present a unified *graphical model* framework for describing codes and deriving iterative decoding algorithms. We illustrate how the following systems can be described using graphical models: turbo-codes, serially-concatenated convolutional codes, frame-oriented turbo-codes, low-density parity-check codes, product codes, and convolutional codes on channels with memory. Recently proposed iterative decoding algorithms (*e.g.*, turbo-decoding) can be viewed as a simple message passing procedure on the graphical model. This framework provides the means to derive new iterative decoders for new codes quite easily.

1 Introduction

In this paper, we seek to unify various themes that underlie much of the recent work on iterative decoding and turbo-codes. We observe that the basic computational paradigm underlying these recently developed techniques appears to be "probability propagation in graphical models," a technique developed in the past decade in the expert systems literature, most notably by Pearl and his co-workers (see *e.g.*, [1]). Interestingly, Gallager's algorithm [2] for decoding low-density parity-check codes is an example of an iterative decoding algorithm that also is essentially an instance of "probability propagation" published over 30 years ago, but mostly forgotten in the interim. Recently, Wiberg, Loeliger and Kötter [3] have refocussed attention on graphical models for codes. They show that such graphical models provide a natural setting in which to describe and study iterative decoding techniques, much as the code trellis is an appropriate model in which to describe and study "conventional" maximum-likelihood soft-decision decoding using the Viterbi algorithm. MacKay and Neal in [4] have also observed that iterative decoding is also essentially an instance of probability or "belief" propagation in a graphical model.

To motivate the general computational paradigm, we first present a simple example of message passing in a "spy network." Although simplistic, this example illustrates many of the important properties of probability propagation algorithms. Next, we review graphical code models, including Markov random fields, Tanner graphs, and Bayesian networks. These graphs have the basic Markov property that only a local conditional probability update is needed to articulate the state of a sub-network.

We then describe the probability propagation algorithm. In the case of singly-connected graphs, probability propagation is an exact algorithm that allows one to compute the *a posteriori* probability of any random variable in the graph, given what has been observed. When the graph has cycles, iterative probability propagation results, with

no guarantee of convergence. However, in some cases (such as turbo-decoding) empirical results have shown great promise for the algorithm.

Finally, we discuss various applications of these results for the decoding of "compound codes," which we think of as codes composed of *constituent codes*, each of which can be decoded tractably by probability propagation on a subgraph of the graph that describes the compound code. Examples of compound codes include turbo-codes [5], serially concatenated convolutional codes [6], frame-oriented turbo-codes [7], low-density parity-check codes [2], and iteratively decoded product codes [8, 9].

2 A Simple "Spy Network"

We now give a simple example of a distributed algorithm that computes a result via local message passing. This example, though simplistic, serves to illustrate many of the important properties of probability propagation algorithms.

In the tradition of spy novels, imagine a "spy network" (or corporate hierarchy) arranged in a tree, like the one shown in Fig. 1(a). Label the root vertex with 'HQ', for "headquarters." The subordinates of a given vertex v (or "controller") is the set of vertices reached by a directed edge from v. In a spy network, a vertex and its subordinates form a "cell" and only the controller has knowledge of the number vertices ("spies") in a given cell.

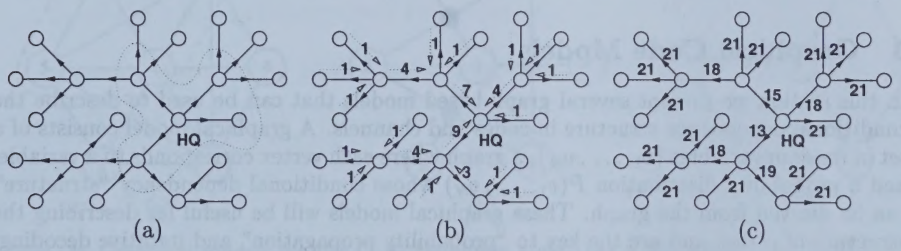


Fig. 1: Message passing in a "spy network." (a) Arrows are directed from controllers to subordinates. (b) Head counts are combined at controller vertices and propagate to the "controller's controller." (c) All members of the network receive the total head count by further message propagation.

We generalize Pearl's distributed soldier counting example [1, p. 155] in the following way. Suppose that HQ wishes to obtain a head count for the entire network. An obvious way that HQ could accomplish this, even without knowledge of the network topology, would be by "delegating" responsibility to the subordinates, each of whom could delegate responsibility to their subordinates, and so on, until "count requests" have propagated as indicated by the arrows of Fig. 1(a) to the leaf vertices.

After receiving head counts from her subordinates, each controller would then sum them, including a count for herself. This head count would then be passed "up" to the controller's controller, and so on, until the head count messages arrive at HQ, who would then be able to ascertain the total network head count (see Fig. 1(b)).

Indeed, every network vertex — not just HQ — could be apprised of the total head count by a further round of message propagation. One approach would be to let each

controller (starting with HQ) inform each subordinate of the total number of vertices in the network that lie in the complement of the network accounted for by that subordinate. In other words, each controller would "articulate" the state of the network for each subordinate, as shown in Fig. 1(c).

In general, in a singly-connected graph, the role of a vertex v is to articulate for a neighbour w the "state" of the sub-network $N_{v\mathbf{A}w}$ reached from w through v, as shown in Fig. 2. Reciprocally, w articulates for v the "state" of the sub-network $N_{w\mathbf{A}v}$. (The notation $N_{v\mathbf{A}w}$ is to be read "the network that v articulates for w.") In statistical terms, v must articulate for w a sufficient statistic, which essentially turns out to be a conditional probability in the case of probability propagation in singly-connected Bayesian networks.

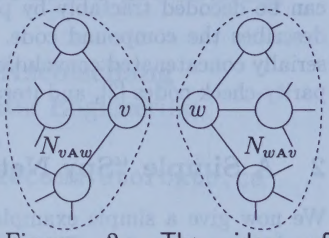


Figure 2. The idea of articulation.

Notice that the graph of Fig. 1 is singly-connected, *i.e.*, the graph has no *undirected cycles* (*i.e.*, cycles where the edge directions are ignored). In contrast, suppose some vertex v is a direct subordinate of its controller's controller. It is quite clear that the message passing algorithm, as described here, would not yield valid results in the presence of such undirected cycles. The controller of v's controller will double-count vertex v's head count because of the direct and indirect paths. This is the sort of problem that makes iterative decoding inexact.

3 Graphical Code Models

In this section we present several graph-based models that can be used to describe the conditional dependence structure in codes and channels. A graphical model consists of a set of random variables $\{v_1, \ldots, v_N\}$, a graph where each vertex corresponds to a variable, and a probability distribution $P(v_1, \ldots, v_N)$ whose conditional dependency "structure" can be derived from the graph. These graphical models will be useful for describing the structure of codes, and are the key to "probability propagation" and iterative decoding.

3.1 Markov Random Fields

Markov random fields (MRFs) are graphical models based on undirected graphs (see textbook reference [10]). If $n(v_i)$ are the neighbors of vertex v_i, then the the distribution $P(\cdot)$ must satisfy the local Markov property:

$$p(v_i|\{v_j\}_{\forall j \neq i}) = p(v_i|n(v_i)). \tag{1}$$

In a MRF, each variable v_i is independent of non-neighbouring vertices in the graph, given the values of its immediate neighbours. The joint distribution is commonly expressed in terms of a Gibbs *potential function* defined on the maximal cliques (*i.e.*, maximal complete subgraphs). Corresponding to each clique q in the set of maximal cliques Q is a collection of vertices V_q. Given a nonnegative potential function $\psi_q(\{v_i \in V_q\})$ for each clique $q \in Q$, the joint distribution over $\{v_1, \ldots, v_N\}$ is given by

$$P(v_1, \ldots, v_N) = Z^{-1} \prod_{q \in Q} \psi_q(\{v_i \in V_q\}), \tag{2}$$

where Z^{-1} is a normalizing constant.

To illustrate how MRFs can be used to describe codes, consider the MRF with seven binary variables shown in Fig. 3a. There are four maximal cliques: $q_1 = \{v_1, v_2, v_5, v_6\}$ (dashed loop), $q_2 = \{v_1, v_4, v_6, v_7\}$, $q_3 = \{v_1, v_3, v_5, v_7\}$, and $q_4 = \{v_1, v_5, v_6, v_7\}$. From (2), $P(v_1, \ldots, v_7) = Z^{-1}\psi_{q_1}(v_1, v_2, v_5, v_6)\ \psi_{q_2}(v_1, v_4, v_6, v_7)\ \psi_{q_3}(v_1, v_3, v_5, v_7)\ \psi_{q_4}(v_1, v_5, v_6, v_7)$. This MRF can be used to describe a Hamming code by setting $\psi_{q_4} = 1$ and by letting the first three potentials be even parity indicator functions. That is, $\psi_q(x_1, x_2, x_3, x_4) = 1$ if its variables have even parity and 0 otherwise. The MRF places equal probability on all codewords that satisfy even parity in cliques q_1, q_2 and q_3.

3.2 Tanner Graphs

Tanner graphs were introduced in [11] for the construction of good long error-correcting codes in terms of shorter codes. Our treatment follows the slightly different presentation of Wiberg, *et. al.* [3]. A Tanner graph is a *bipartite* graph representation for a check structure, similar to the one described above. In such a graph, there are two types of vertices corresponding to the variables and the "checks", respectively, with no edges connecting vertices of the same type. For example, a Tanner graph corresponding to the Hamming code described above is shown in Fig. 3b. Each check vertex q in the set of check vertices Q is shown as a filled circle. A check vertex ensures that its set of neighbours satisfies even parity in a valid configuration.

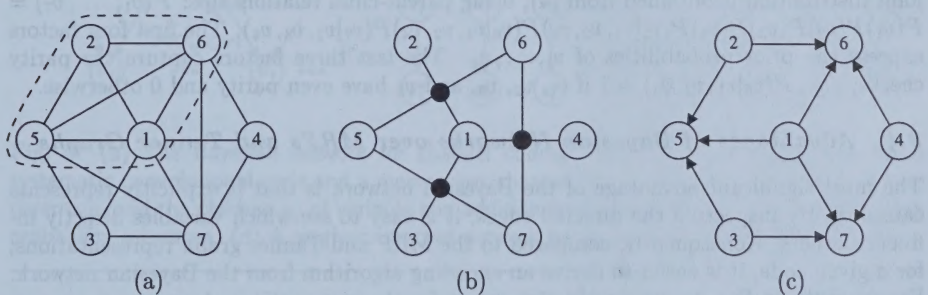


Fig. 3: Graphical models for the (7,4) Hamming code: (a) a Markov random field, with a maximal clique indicated; (b) a Tanner graph; and (c) a Bayesian network.

We see that the check vertices have a similar role in a Tanner graph as the maximal cliques have in a Markov random field. In general, for each check vertex q with neighbours $n(q)$, we can associate a non-negative real-valued potential function $\psi_q(\{v_i \in n(q)\})$. We then write a probability distribution over the variables as

$$P(v_1, \ldots, v_N) = Z^{-1} \prod_{q \in Q} \psi_q(\{v_i \in n(q)\}), \tag{3}$$

where Z^{-1} is a normalizing constant. Of course, (3) is analogous to (2).

An MRF can be converted to a Tanner graph by introducing a check vertex for each maximal clique, with edges connecting that check vertex to each variable in the clique. A Tanner graph can be converted to an MRF by eliminating the check vertices and forming cliques from all variables originally connected to the same check vertex. However, Tanner graphs may be more specific about dependencies than MRFs. Different Tanner graphs may map to the same MRF; *e.g.*, the graph in Fig. 3b with an additional check vertex connected to v_1, v_5, v_6 and v_7 will also map to the MRF in Fig. 3a.

3.3 *Bayesian Networks*

We now introduce Bayesian networks that, unlike Markov random fields and Tanner graphs, have *directed acyclic* graphical representations [1]. A directed acyclic graph is one where there are no graph cycles when the edge directions are followed. (There may be cycles when the edge directions are ignored.) Let the *parents* $a(v_i)$ of vertex v_i be the set of vertices that have directed edges connecting *to* v_i. For a Bayesian network, the joint distribution can be written

$$P(v_1, \ldots, v_N) = \prod_{i=1}^{N} P(v_i | a(v_i)), \tag{4}$$

where, if $a(v_i) = \varnothing$ (*i.e.*, v_i has no parents) then we take $p(v_i|\varnothing) = p(v_i)$. Every distribution can be described by a Bayesian network, since by the chain rule, $P(v_1, \ldots, v_N) = P(v_1)P(v_2|v_1)P(v_3|v_1, v_2) \ldots P(v_N|v_1, v_2, \ldots, v_{N-1})$. So, we can pick any ordering of the variables and then condition each variable on all previous variables. However, this trivial network does not capture any useful probabilistic structure. The last factor, $P(v_N|v_1, v_2, \ldots, v_{N-1})$, contains all N variables and so is really just as complicated as the full joint distribution.

A Bayesian network for the Hamming code described above is shown in Fig. 3c. The joint distribution is obtained from (4), using parent-child relationships: $P(v_1, \ldots, v_7) = P(v_1)P(v_2)P(v_3)P(v_4)P(v_5|v_1, v_2, v_3)P(v_6|v_1, v_2, v_4)P(v_7|v_1, v_3, v_4)$. The first four factors express the prior probabilities of $v_1, \ldots, v_4$. The last three factors capture the parity checks; *e.g.*, $P(v_5|v_1, v_2, v_3) = 1$ if v_1, v_2, v_3, and v_5 have even parity and 0 otherwise.

3.4 *Advantages of Bayesian Networks over MRFs and Tanner Graphs*

The most significant advantage of the Bayesian network is that it explicitly represents causality. By inspecting the directed edges, it is easy to see which variables directly influence others. Consequently, compared to the MRF and Tanner graph representations, for a given code, it is easier to derive an encoding algorithm from the Bayesian network. For example, in Fig. 3c, we simply pick values for the systematic root vertices v_1, v_2, v_3, and v_4 and then determine the remainder of the codeword, v_5, v_6, and v_7. The MRF and Tanner graphs do not specify such a temporal construction for the code. The explicit representation of causality is also useful for modelling physical effects, such as channel noise. For example, in a channel with memory, the current received signal depends on the corresponding transmitted codeword symbol *and earlier ones.* Whereas the MRF and Tanner graph cannot explicitly represent this temporal causality, the Bayesian network can (as we shall see later). As described in the next section, for channel coding applications, it will usually be convenient to arrange the variables roughly in the order of input, state, codeword, channel output.

3.5 *Bayesian Networks for Channel Coding*

In coding, the relationships between the information symbols $\mathbf{u}$, the encoder state variables $\mathbf{s}$ (if there are any), the transmitted codeword symbols $\mathbf{x}$, and the received signals $\mathbf{y}$ completely define the encoding and decoding problem for a given code. The Bayesian network for channel coding in general is shown in Fig. 4a. By inspection of the network, the joint distribution is $P(\mathbf{u}, \mathbf{s}, \mathbf{x}, \mathbf{y}) = P(\mathbf{u})P(\mathbf{s}|\mathbf{u})P(\mathbf{x}|\mathbf{u}, \mathbf{s})P(\mathbf{y}|\mathbf{x})$. Usually, $P(\mathbf{u})$ is a uniform distribution and $P(\mathbf{s}|\mathbf{u})$ and $P(\mathbf{x}|\mathbf{u}, \mathbf{s})$ are deterministic (*i.e.*, all probability mass is placed on a single outcome). The *channel likelihood* $P(\mathbf{y}|\mathbf{x})$ expresses the noise and inter-symbol interference introduced by the channel.

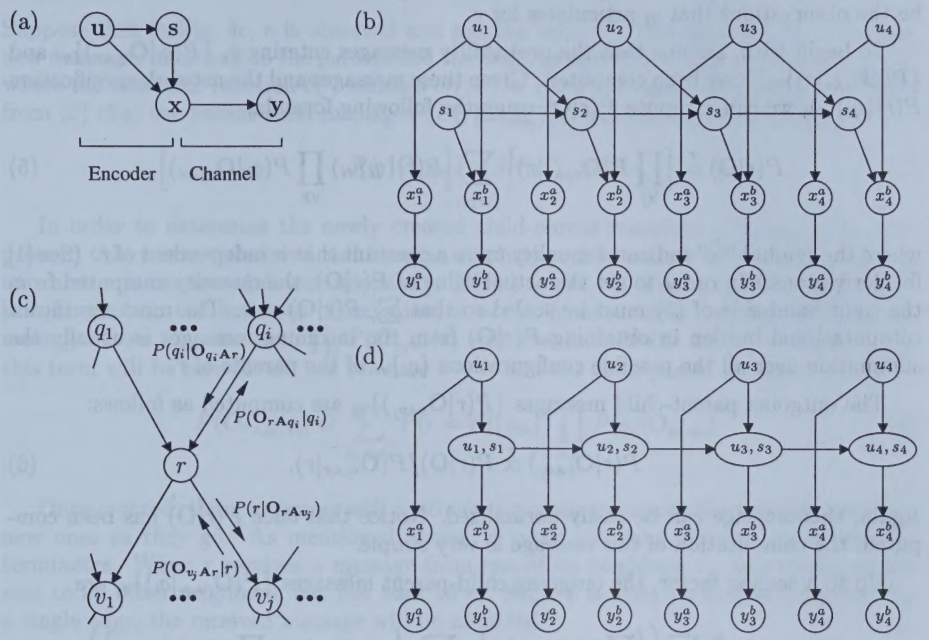


Fig. 4: (a) The Bayesian network for channel coding. (b) The Bayesian network for a systematic convolutional code and a memoryless channel. (c) A network fragment shows the parents q_i and the children v_j of variable r in addition to some of the incoming and outgoing probability messages. (d) A singly-connected network for the convolutional code from (b).

Fig. 4b shows the Bayesian network for a systematic convolutional code with a memoryless channel. The systematic codeword symbols are denoted by "a"; the symbols denoted by "b" are outputs of the convolver, where x_k^b depends on u_k *and* state s_k. By inspecting the parents of the received signals, we find that $P(\mathbf{y}|\mathbf{x}) = \prod_k P(y_k^a|x_k^a)P(y_k^b|x_k^b)$, which expresses the absence of memory in the channel.

4 Probability Propagation

In this next section, we describe the basic probability propagation algorithm that forms the basis for iterative decoding. As an example of how probability propagation is useful, suppose we observe the channel output in the Bayesian network for a code; we would then like to compute the *a posteriori* information symbol probabilities in order to perform MAP decoding. Probability propagation is a method of propagating the effects of one or more observations across a Bayesian network so that each unobserved node can compute the probability for its variable *conditioned* on the observed variables.

Fig. 4c shows a network fragment consisting of a variable r, its parents $\{q_i\}_{\forall i}$, and its children $\{v_j\}_{\forall j}$. Some of the other variables not shown have been observed and we wish to determine the effect of these observations on the probability of r; that is, we want to compute $P(r|\mathbf{O})$ where $\mathbf{O}$ is the set of observations. Since the network is singly-connected, each parent and child of r articulates the observations in the respective

subnetwork. Let $\mathbf{O}_{v_j \mathbf{A} r}$ be the observations that v_j articulates for r. Similarly, let $\mathbf{O}_{q_i \mathbf{A} r}$ be the observations that q_i articulates for r.

To begin with, assume that the probability messages entering r, $\{P(q_i|\mathbf{O}_{q_i \mathbf{A} r})\}_{\forall i}$ and $\{P(\mathbf{O}_{v_j \mathbf{A} r}|r)\}_{\forall j}$, have been computed. Given these messages and the network specification $P(r|\{q_i\}_{\forall i})$, we can compute $P(r|\mathbf{O})$ using the following formula:

$$P(r|\mathbf{O}) \overset{r}{\propto} \Big[\prod_{\forall j} P(\mathbf{O}_{v_j \mathbf{A} r}|r)\Big] \sum_{\{q_i\}_{\forall i}} \Big[P(r|\{q_i\}_{\forall i}) \prod_{\forall k} P(q_k|\mathbf{O}_{q_k \mathbf{A} r})\Big], \tag{5}$$

where the symbol "$\overset{r}{\propto}$" indicates equality up to a constant that is independent of r. (See [1] for derivations.) In order to get the actual value of $P(r|\mathbf{O})$, the quantity computed from the right hand side of (5) must be scaled so that $\sum_r P(r|\mathbf{O}) = 1$. The most significant computational burden in obtaining $P(r|\mathbf{O})$ from the incoming messages is usually the summation over all the possible configurations $\{q_i\}_{\forall i}$ of the parents of r.

The outgoing parent-child messages $\{P(r|\mathbf{O}_{r \mathbf{A} v_j})\}_{\forall j}$ are computed as follows:

$$P(r|\mathbf{O}_{r \mathbf{A} v_j}) \overset{r}{\propto} P(r|\mathbf{O}) / P(\mathbf{O}_{v_j \mathbf{A} r}|r). \tag{6}$$

Again, this message can be easily normalized. Notice that once $P(r|\mathbf{O})$ has been computed, the computation of this message is very simple.

Up to a scaling factor, the outgoing child-parent messages $\{P(\mathbf{O}_{r \mathbf{A} q_i}|q_i)\}_{\forall i}$ are

$$P(\mathbf{O}_{r \mathbf{A} q_i}|q_i) \overset{q_i}{\propto} \sum_r \Bigg(\Big[\prod_{\forall j} P(\mathbf{O}_{v_j \mathbf{A} r}|r)\Big] \sum_{\{q_k\}_{\forall k \neq i}} \Big[P(r|\{q_k\}_{\forall k}) \prod_{\forall l \neq i} P(q_l|\mathbf{O}_{q_l \mathbf{A} r})\Big] \Bigg). \tag{7}$$

In this case, since q_i is the condition for $P(\mathbf{O}_{r \mathbf{A} q_i}|q_i)$, this probability can't be easily normalized. (To normalize it would require a summation over all the possible observation configurations $\mathbf{O}_{r \mathbf{A} q_i}$ that r articulates for q_i.) However, it turns out that these child-parent messages can be used in un-normalized form. They are used to compute (5) and (6), which can be normalized.

These update equations provide a consistent way to propagate conditional probability messages across a singly-connected network. When the network is *not* singly-connected, the propagation rules can still be applied, but there is no guarantee as to the quality of the results. Soon, we shall see that this is exactly what turbo-decoding in its simplest form does.

4.1 Buffered Messages

The above update rules do not prescribe a *schedule* for passing the messages. When r receives a message from one of its children, must r immediately and in parallel propagate messages to its other children and its parents? It turns out that when the network is singly-connected, the propagation schedule does not matter, as long as all of the messages are eventually *absorbed*, as described below. Consequently, it is often useful to buffer messages. For example, if we know that r will soon receive messages from all the other children, we can pass fewer messages in total by waiting for those others to arrive before applying the update rules (5), (6) and (7). Pictorially, when a message arrives on an edge, but is not propagated on to the other neighbors, we draw a small dot on each of the other edges. These dots can be turned into arrows (indicating a message is being passed) at any time, as shown in Fig. 5.

4.2 *Message Creation and Absorption*

Suppose that in Fig. 4c, r is observed and has the value r'. This observation will cause new messages to be sent to the parents and the children of r. Obviously, $P(r|\mathbf{O}) = \delta(r, r')$, where the indicator function $\delta(\cdot)$ satisfies $\delta(r, r') = 1$ if $r = r'$ and 0 otherwise. It follows from (6) that the parent-child messages $\{P(r|\mathbf{O}_{r\mathbf{A}v_j})\}_{\forall j}$ can be computed from

$$P(r|\mathbf{O}_{r\mathbf{A}v_j}) = \delta(r, r'). \tag{8}$$

In order to determine the newly created child-parent messages $\{P(\mathbf{O}_{r\mathbf{A}q_i}|q_i)\}_{\forall i}$, we pretend that r is not observed and imagine a fictitious child node z that *is* observed and to which r is connected deterministically: $P(z|r) = \delta(z, r)$. The authentic observation is equivalent to the fictitious observation, $z = r'$. When determining the child-parent messages from (7), the term $\prod_j P(\mathbf{O}_{v_j\mathbf{A}r}|r)$ will now include the fictitious child z. So, this term will be zero for all but the case $r = r'$ and the child-parent messages are

$$P(\mathbf{O}_{r\mathbf{A}q_i}|q_i) \overset{q_i}{\propto} \sum_{q_k : k \neq i} P(r = r'|\{q_m\}) \prod_{n \neq i} P(q_n|\mathbf{O}_{q_n\mathbf{A}r}). \tag{9}$$

Once created, these messages will continue to propagate across the network, creating new ones as they go. As mentioned above, it turns out that this process eventually terminates. When r receives a message from one of its neighbors v_j, new messages are sent to all other neighbors, but *not* back to v_j. So, if r is a vertex that is connected by a single edge, the received message will be *absorbed.*

4.3 *Network Initialization*

Before propagating the effects of observations across the network, we must produce initial messages that correspond to the *a priori* status of the network. We obtain the initial child-parent messages by noting that $\mathbf{O} = \varnothing$ and so $P(\mathbf{O}_{r\mathbf{A}q_i}|q_i) = P(\varnothing|q_i) = 1$, for all q_i. Also, $P(r|\mathbf{O}_{r\mathbf{A}v_j}) = P(r|\varnothing) = P(r) = \sum_{\{q_i\}_{\forall i}} P(r|\{q_i\}_{\forall i}) \prod_j P(q_j)$. So, if r is a root node, it simply passes its marginal probability $P(r)$ to its children. If r is not a root node, it uses the marginal probabilities of its parents to compute $P(r)$. These messages propagate from parent to child until each variable has an initial probability.

4.4 *The Forward-Backward (BCJR) Algorithm*

As a familiar example of probability propagation, consider the Bayesian network for a convolutional code. The multiply-connected network shown in Fig. 4b can be converted to the singly-connected network shown in Fig. 4d. By grouping information and state variables together, we eliminate all undirected cycles, at the expense of increasing the complexity of the state variables. The *forward-backward algorithm* (a.k.a. the "BCJR algorithm") for computing $P(u_k|\mathbf{y})$ for such a Markov-type model was first developed by Baum and Petrie [12]. It turns out that this algorithm is identical to the algorithm that results when probability propagation is applied to the corresponding Bayesian network. A simpler, nonsystematic version of the network in Fig. 4d is shown by the first picture in Fig. 5. (We have included the codeword symbols with the state variables, for the sake of graphical simplicity.) When the channel output is observed (crossed circles), messages propagate up to the state vertices. These messages are the codeword symbol likelihoods as determined by the channel model $P(y_k|x_k)$ and the channel output $\mathbf{y}$. In order to minimize the total number of messages passed, the messages are buffered as shown. A series of parent-child messages are passed *forward* along the chain. Then, a series of

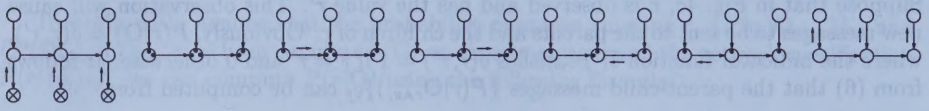


Fig. 5: Probability propagation in the Bayesian network for a convolutional code.

child-parent messages are passed *backward* along the chain. These two types of message are exactly the same as the "α"s and "β"s in the standard forward-backward algorithm. Finally, the information symbols receive probability messages that allow them to compute $P(u_k|\mathbf{y})$.

Although this example does not really improve our understanding of decoding a single convolutional code, it does illustrate how algorithms for decoding on Bayesian networks can be easily derived using the probability propagation idea. In the following section, we will present the Bayesian networks for several recent codes, including turbo-codes, and illustrate that the new iterative decoding algorithms (*e.g.*, turbo-decoding) are instances of probability propagation applied to networks that are *not* singly-connected.

5 Compound Codes

We define a *compound code* to be one which can be described by a set of constituent codes, each of which is tractably decodable on its own. Graphically, each constituent code is represented by a singly-connected constituent Bayesian network. These constituent networks share some variables, so that taken as a whole, the total Bayesian network is not singly-connected.

5.1 Bayesian Networks for Some Known Codes

The Bayesian network for a systematic rate 1/3 compound turbo-code is shown in Fig. 6a, along with its singly-connected constituent networks. This compound code consists of two chain-type networks that are connected using a different ordering of the information symbol vertices. Whereas the upper chain directly uses the information sequence, the lower chain uses a permuted version of the information sequence.

A serially concatenated convolutional compound code [6] is shown in Fig. 6b. This system is essentially the same as Forney's concatenated codes [13], except that a convolutional code is used for both the inner and outer codes, instead of the more popular Reed-Solomon outer code.

A frame-oriented turbo-code [7] is shown in Fig. 6c. This code was designed to eliminate the need for trellis termination when using short block lengths.

Fig. 6d shows a low-density parity check code [2]. This code consists of parity check restrictions (upper vertices) on subsets of the codeword symbols (middle vertices). MacKay and Neal [4] were the first to describe Gallager's codes using Bayesian networks.

Fig. 6e shows a systematic product code. The distribution for the parity symbol in each row (column) is conditioned on the information symbols in its row (column), and is nonzero only for even parity row-wise (column-wise) configurations.

The Bayesian network for a convolutional code on a channel with 1st order memory is shown in Fig. 6f. In this case, the *codeword symbols* are permuted before transmission.

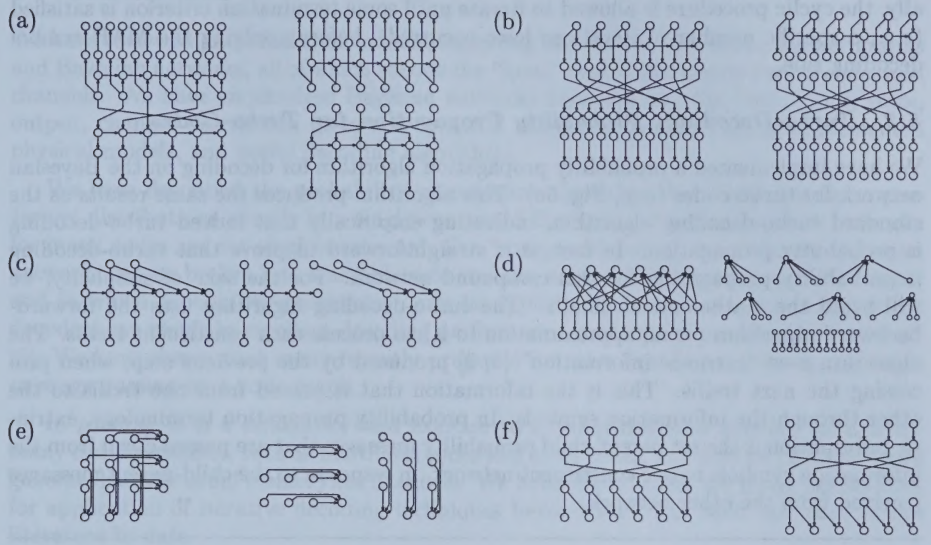


Fig. 6: The Bayesian networks for several coding systems: (a) a systematic turbo-code; (b) a nonsystematic serially concatenated convolutional code; (c) a nonsystematic frame-oriented turbo-code; (d) a low-density parity check code; (e) a product code; and (f) a convolutional code on a channel with 1st order memory, with permuted codeword symbols.

5.2 Soft Iterative Decoding: Probability Propagation for Compound Codes

Since each constituent Bayesian network in a compound code is singly-connected, probabilities for each random variable given the observed random variables can be efficiently computed *exactly* within each constituent network. However, because the compound network is multiply-connected, probability propagation is only an approximate algorithm for computing these probabilities. As we see it, the general idea of iterative decoding is to make use of the efficient probability propagation algorithm for each constituent network, while either ignoring the cycles or somehow approximately taking them into account. This graphical framework unifies several iterative decoding algorithms. The iterative decoding algorithms for turbo-codes, serially concatenated convolutional codes, frame-oriented turbo-codes, low-density parity-check codes, product codes, and inter-symbol interference systems can all be viewed as probability propagation in the networks shown in Fig. 6.

The overall decoding procedure essentially consists of applying probability propagation while ignoring the graph cycles. The procedure is usually be broken down into a series of processing cycles. In each cycle, probabilities are propagated across a particular constituent network, producing current estimates of the distributions over information symbols, state variables and codeword symbols, given the observed channel output. The next cycle then uses the probability estimates produced by the previous cycle when processing the next constituent network. Usually, the constituent codes are processed in order and one pass through all of the codes is called an *iteration*. An iteration essentially consists of propagating probabilities across the network as if it were singly-connected, stopping when each vertex has been processed *once*. In fact, because the compound code

network is multiply-connected, the propagation procedure never self-terminates. Usually, the cyclic procedure is allowed to iterate until some termination criterion is satisfied (*e.g.*, a specific number of iterations have occurred), before applying the MAP symbol decoding rule.

5.3 Turbo-Decoding: Probability Propagation for Turbo-Codes

We have implemented a probability propagation algorithm for decoding on the Bayesian network for turbo codes (*e.g.*, Fig. 6a). This algorithm produces the same results as the standard turbo-decoding algorithm, indicating empirically that indeed turbo-decoding is probability propagation. In fact, it is straightforward to prove that turbo-decoding is probability propagation for this compound network. For the sake of simplicity, we will avoid the mathematical details. The turbo-decoding algorithm uses the forward-backward algorithm (or an approximation to it) to process each constituent trellis. The algorithm uses "extrinsic information" [5, 9] produced by the previous step, when processing the next trellis. This is the information that is passed from one trellis to the other through the information symbols. In probability propagation terminology, extrinsic information is the set parent-child probability messages that are passed down from the information symbols to one constituent network, in response to the child-parent messages received from the other network.

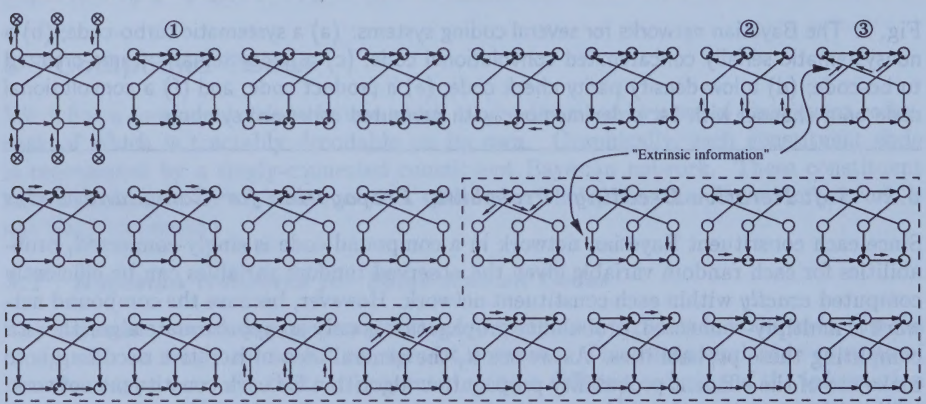


Fig. 7: Probability propagation in the Bayesian network for a turbo-code.

Fig. 7 shows the message passing dynamics for a simplified turbo-code Bayesian network. When the channel output is observed (as shown by the crossed circles), messages propagate up to the state vertices of both constituent networks. One constituent network is processed in the manner of the forward backward algorithm, ① to ②. The information symbol vertices receive probability messages from the constituent network just processed. In the case of the single code (Fig. 5), the information symbols are each connected by a single edge and so propagation terminates. In this case, however, a set of messages are then passed to the other constituent network ③ (these messages are the "extrinsic information"). The overall procedure is cyclical and repeats over and over as shown by the pictures outlined by a dashed polygon.

6 Conclusions

We have reviewed graphical code models, including Markov random fields, Tanner graphs and Bayesian networks, all of which encode the "local" probabilistic structure of codes and channels. We have emphasized Bayesian networks here, because the basic input, state, output, channel description gives rise to useful encoder structures, easily implemented physical models, and useful decoding algorithms.

We have described the probability propagation algorithm that computes the *a posteriori* distributions *exactly* in a singly-connected Bayesian network. Bayesian networks for many codes are not singly-connected. For "compound codes," however, the Bayesian networks can be broken into tractable subnetworks, each describing a "constituent code" and in which probability propagation can be applied. Iterating over these constituent decoders can result in excellent decoding performance, as demonstrated by Berrou, *et al.* [5]. We have shown that many recently proposed iterative decoders can be described as message passing on a code graph.

In general, it is a straightforward exercise to develop Bayesian network models for many coding schemes, such as multilevel codes and coset codes, and also for channels more general than the usual memoryless channels. We believe that there are many possibilities for application of iterative decoding techniques beyond what has been described in the literature to date.

We appreciate the discussions we had with Glenn Gulak, David MacKay, Radford Neal and Niclas Wiberg.

References

[1] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference.* San Francisco, CA: Morgan Kaufmann, 1988. Revised second printing.

[2] R. G. Gallager, *Low-Density Parity-Check Codes.* Cambridge, MA: M.I.T. Press, 1963.

[3] N. Wiberg, H.-A. Loeliger, and R. Kötter, "Codes and iterative decoding on general graphs," *European Trans. on Telecommun.*, vol. 6, pp. 513–525, Sep./Oct. 1995.

[4] D. J. C. MacKay and R. M. Neal, "Good codes based on very sparse matrices." Submitted to *IEEE Transactions on Information Theory*, 1996.

[5] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon limit error-correcting coding and decoding: Turbo codes," in *Proc. IEEE Int. Conf. Commun. (ICC)*, (Geneva, Switzerland), pp. 1064–1070, 1993.

[6] S. Benedetto and G. Montorsi, "Iterative decoding of serially concatenated convolutional codes," *Electronics Letters*, vol. 32, pp. 1186–1188, 1996.

[7] C. Berrou and M. Jezequel, "Frame-oriented convolutional turbo-codes," *Electronics Letters*, vol. 32, pp. 1362–1364, 1996.

[8] J. Lodge, R. Young, P. Hoeher, and J. Hagenauer, "Separable MAP 'filters' for the decoding of product and concatenated codes," in *Proceedings of IEEE International Conference on Communications*, pp. 1740–1745, 1993.

[9] J. Hagenauer, E. Offer, and L. Papke, "Iterative decoding of binary block and convolutional codes," *IEEE Transactions on Information Theory*, vol. 42, pp. 429–445, 1996.

[10] R. Kindermann and J. L. Snell, *Markov Random Fields and their Applications.* Providence, Rhode Island: American Mathematical Society, 1980.

[11] R. M. Tanner, "A recursive approach to low complexity codes," *IEEE Trans. on Inform. Theory*, vol. IT-27, pp. 533–547, Sept. 1981.

[12] L. E. Baum and T. Petrie, "Statistical inference for probabilistic functions of finite state markov chains," *Annals of Mathematical Statistics*, vol. 37, pp. 1559–1563, 1966.

[13] G. D. Forney, Jr., *Concatenated Codes.* Cambridge MA.: MIT Press, 1966.

Some High-Rate Near Capacity Codecs for the Gaussian Channel

Jung-Fu Cheng
jfc@systems.caltech.edu

Robert J. McEliece
rjm@systems.caltech.edu

WWW Address: http://www.cco.caltech.edu/~rjm/
California Institute of Technology 136-93, Pasadena, CA 91125, USA

Abstract

We present a construction of high-rate linear error-correcting codes based on random bipartite graphs. In contrast to other researchers, our approach uses these graphs to produce generator matrices. A distributive parallel decoding algorithm based on the principle of belief propagation is presented. These new codes are then shown to be linear-time encodable and decodable. Experimental results further show the performances of these new systems to be approach the channel capacities of the Gaussian channels with binary or multilevel signal constellations.

1 Introduction

The introduction of multi-stage iterative decoding with exchange of soft information [3] has recently attracted substantial research interests and reperesents a genuine breakthrough in coding theory. This methodology has been applied to parallel concatenated convolutional (turbo) codes [5, 11, 12, 17, 22], parallel concatenated block codes [17], serial concatenated convolutional codes [2], and generalized concatenated convolutional codes [6,7,14] with remarkable success. Most of these reports focus on experimental simulation results with little on explanatory theory for the code structure [1,12,13] and even less on theoretical examination for the decoding mechanism [22]. However, a close connection between the turbo decoding algorithm and the "belief propagation" algorithm of artificial intelligence [23] has recently been found [21]. There it is shown that, by representing a turbo code with a "Markov field network" as one in Fig. 1 for a classical turbo code, the application of the belief propagation algorithm directly leads to the turbo decoding algorithm. (A Markov field network is directed graph describing the interdependence among random variables of a stochastic system. For the example in Fig. 1, the white nodes represents input data bits and the directed edges from them to the two black

*This work was supported by NSF grant NCR-9505975.

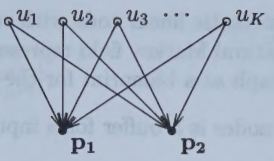


Figure 1: A Markov field representation of a classical turbo code.

nodes for parity blocks describe that the later are determined by the former in the encoding process.) This framework also prescribes explicitly how the decoders for multiple turbo codes [11] should proceed without any *ad hoc* arguments. In fact, it is also argued that the decoding algorithms for all the aforementioned variations of the turbo coding construction as well as a wide range of low-density parity check codes [15,16,19,20,26,28] can be explained by this principle.

It is the purpose of this paper to further examine the applicability of the belief propagation algorithm to decode error-correcting codes. It seems unnecessary to separate the parity bits into two subsets as the case of classical turbo codes shown in Fig. 1. In Section 2, we shall propose a construction of systematic binary linear codes which treats all the parity bits equally. A distributive parallel decoding algorithm based on the principle of belief propagation is presented in Section 3. Experimental results are then presented in Section 3. Finally, Section 4 contains our concluding remarks.

2 Construction of the Code

The first step is to construct a random regular [1] bipartite graph $\mathcal{B}_{KS}$ with the size of the two partite sets as K and S. Call these two subsets the *systematic* nodes and *parity* nodes, and label the vertices in them by $u_1, u_2, \ldots, u_K$ and $p_1, p_2, \ldots, p_S$, respectively, as shown in Fig. 2. The degree of each of the systematic nodes is required to be χ and hence the degree of each of the parity nodes is $\frac{K}{S}\chi$. The construction of such a graph can be accomplished either by trial-and-error with a computer program or by combinatorial methods [9].

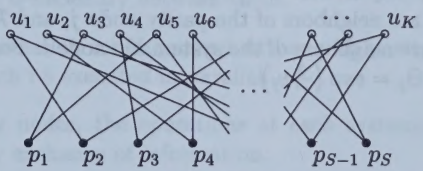


Figure 2: A random regular bipartite graph.

Let $G = [I|P]$ be the generator matrix of the new code, where I is a $K \times K$ identity matrix and P is a $K \times S$ matrix. The entry P_{ij} of the matrix P will be 1 if and only if there is an edge connecting u_i and p_j, and 0 otherwise. This construction thus results in a K

[1] The regularity of the graph is not essential for our decoding algorithm, but, in practice, it could lead to simplified modular hardware implementation.

input, $N = K + S$ output systematic linear code with random parity bits. Furthermore, the bipartite graph forms a natural Markov field representation of the new code. We can also interpret the bipartite graph as a blueprint for the encoder in the following way:

1. Each of the systematic nodes is a buffer for a input information bit.
2. All the edges are directed from the systematic nodes towards the parity nodes.
3. Each of the parity nodes is in fact a modulo 2 adder.

Since χ is independent of K, the total encoding complexity $K\chi$ is, therefore, linear in the information length K.

3 System Model and the Decoding Algorithm

The input information bits to the encoder are assumed to take binary values equiprobably. The encoded bits are mapped into ± 1 ($0 \to +1$ and $1 \to -1$) and then sent over an additive white Gaussian channel. That is, the received signals are assumed to be the sum the transmitted signals and independent white Gaussian random variables with zero mean and variance $\sigma^2 = N_0/2E_s = N_0/2RE_b$, where E_s is the average energy per symbol, E_b the average energy per bit, R the code rate, and N_0 the single-sided noise power spectral density. Denote the received signals corresponding to the systematic bits by x_i, $(i = 1, 2, \ldots, K)$, and the received signals corresponding to the parity bits by y_j, $(j = 1, 2, \ldots, S)$.

3.1 The Decoding Algorithm

Just as the encoding process is based on the bipartite graph, the decoder is also matched to the graph by an application of Pearl's belief propagation algorithm [23]. It can also be viewed as a generator domain counterpart of those algorithms appeared before [15, 16, 19, 20, 26, 28], whose operations are based on the parity check matrices.

Let $\mathcal{U} = \{1, 2, \ldots, K\}$ be the index set for the systematic nodes, and $\mathcal{P} = \{1, 2, \ldots, S\}$ the index set for the parity nodes. Furthermore, let $\mathcal{U}(j) \subset \mathcal{U}$ be the index set of the systematic nodes that are neighbors of the parity node j, and $\mathcal{P}(i) \subset \mathcal{P}$ the index set of the parity nodes that are neighbors of the systematic node i. For $i \in \mathcal{U}$, define $\phi_i = \frac{2}{\sigma^2} x_i$, and, for $j \in \mathcal{P}$, define $\Theta_j = \exp\left(\frac{2}{\sigma^2} y_j\right)$.

✎ **Initialization**

```
1  for i ∈ U do
2      B_i := φ_i;
3      for j ∈ P do
4          L_ji := 0;
5      od
6  od
```

✎ **Updating Rules for Parity Nodes**

7 **for** $j \in \mathcal{P}$ **do**
8 **for** $i \in \mathcal{U}(j)$ **do**
9 $\delta_{ji} := \frac{\exp(B_i - L_{ji}) - 1}{\exp(B_i - L_{ji}) + 1}$;
10 **od**
11 $D_j := \prod_{i \in \mathcal{U}(j)} \delta_{ji}$;
12 **for** $i \in \mathcal{U}(j)$ **do**
13 $\triangle := D_j / \delta_{ji}$;
14 $L_{ji} := \log \frac{(1+\triangle)\Theta_j + (1-\triangle)}{(1-\triangle)\Theta_j + (1+\triangle)}$;
15 **od**
16 **od**

✎ **Updating Rule for Systematic Nodes**

17 **for** $i \in \mathcal{U}$ **do**
18 $B_i := \phi_i + \sum_{j \in \mathcal{P}(i)} L_{ji}$;
19 **od**

✎ **Decision Phase**

20 **for** $i \in \mathcal{U}$ **do**
21 $\hat{u}_i := \begin{cases} 0, & \text{if } B_i \geq 0 \\ 1, & \text{if } B_i < 0 \end{cases}$;
22 **od**

3.2 Comments about the Algorithm

The most significant characteristic of the proposed algorithm is its highly distributive parallelism, which contrasts to the turbo decoding paradigms where sequential processing is accomplished by very powerful central units [2,3,5,11,17]. More specifically,

- The operations in steps 8–15 can be executed locally at each of the parity nodes. No communication is necessary between them.
- Further parallel processing could be done within each parity node: steps 8–10 and steps 12–15 can each be executed in parallel.
- As with the parity nodes, the operations at each systematic node can be done locally without any exchange of information.

A direct implementation of the algorithm would require, for each iteration, about $5\frac{K}{S}\chi$ multiplications/divisions (MUL/DIV) and $3\frac{K}{S}\chi$ additions/subtractions (ADD/SUB) for each of the parity nodes, and $\frac{K}{S}\chi$ ADD/SUB for each of systematic nodes. The total decoding complexity is thus about $5K\chi$ MUL/DIV and $4K\chi$ ADD/SUB, which is linear in the information length K. The complexity, however, can be greatly reduced as follows:

- The complex operation in step 9 can be avoided by approximating the function with a piece-wise linear function.

- The number of computations in step 14 can be reduced by noticing that

$$L_{ji} \to \begin{cases} \pm\frac{2}{\sigma^2}y_j, & \text{if } \triangle \to \pm 1 \\ \pm \log\frac{1+\triangle}{1-\triangle}, & \text{if } \pm\frac{2}{\sigma^2}y_j \gg 0 \end{cases}.$$

- Though many MUL/DIV are used in the above algorithm, for most of the cases, they can be replaced by ADD/SUB by converting the appropriate variables into the log domain. Since these considerations depend on specific hardware implementation, details will not be further explored here.

4 Simulation Results

The performances of two $(1536, 1024)$ codes with $\chi = 6$ and $\chi = 7$ are shown in Fig. 3. It is observed that iterative decoding improves the performance by a large amount. For example, at $E_b/N_0 = 3.5$ dB, the bit error rate (BER) of the systems with $\chi = 7$ is 2×10^{-2} after the first iteration. It is decreased to 8×10^{-5} after four iterations and further reduced to 1.6×10^{-6} after 25 iterations. However, it is also observed from these plots that about ten iterations are enough to obtain most of the performance gains. At BER$= 10^{-5}$, these two coding systems offer gains of more that 6.5 dB over an uncoded system, and, in fact, their performances are within 2 dB of the channel capacity with rate $2/3$ and binary signaling. These impressive performance improvements are achieved with a relatively short block length of 1024. We expect larger coding gains can be achieved with longer block lengths.

An "error floor" phenomenon similar to those reported for the turbo codes [12] can be observed in the performance curves for the $\chi = 6$ code. It has been argued the "floor"

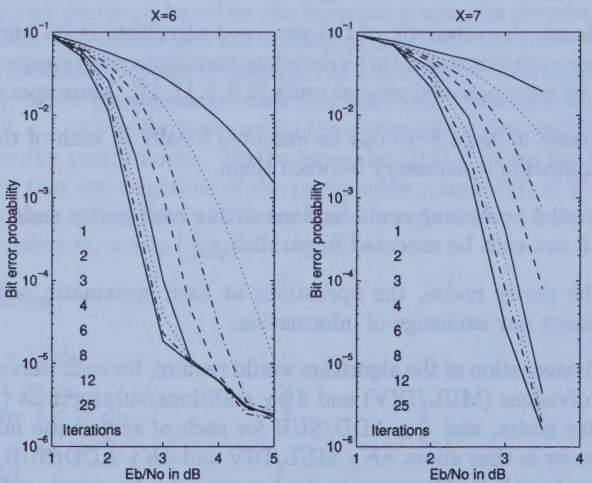


Figure 3: Performances of the $(1536, 1024)$ systematic random linear codes with $\chi = 6$ and $\chi = 7$.

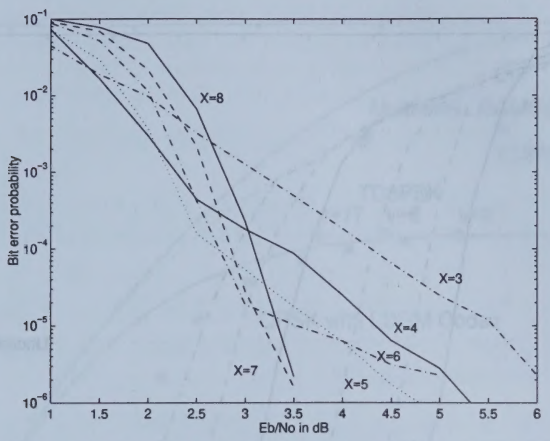


Figure 4: Performances of the (1536, 1024) systematic random linear codes with $\chi = 3, 4, 5, 6, 7,$ and 8 after 16 iterations.

is caused by the true minimum distance of the code. Since no such floor is observed for the $\chi = 7$ code for BER down to 10^{-6}, it seems to be a better code than the $\chi = 6$ one in the sense that it has larger minimum distance or smaller number of nearest neighbor codewords. As shown in Fig. 4, however, the performance of the $\chi = 8$ code is inferior to that of the $\chi = 7$ code. Though there is no guarantee that the $\chi = 8$ code is better, it is suspected that the proposed algorithm might not be able to extract all the power of codes with small cycles such as the $\chi = 8$ code. This might be because of the smaller sizes of girths in the bipartite graphs [15,16,19,20] with this high connection parameters.

More high-rate codes with the parameters listed in Table 1 have been constructed and studied. The performance curves of the codes with $K = 1024$ after 16 iterations of decoding are shown in Fig. 5. Quite impressively, with only 3% redundancy, the (1056, 1024) code achieves a coding gain of 3.5 dB over an uncoded system at BER $= 10^{-6}$. In addition, with a relatively short block length of 1024, the performances of these codes are all within about 2 dB of the channel capacities for their corresponding rates. Another distinguishing characteristic of the proposed decoding algorithm is its ability to base its decisions more on the systematic bits than on the parity bits when the

K	S	Rate R	χ
1024	16	0.985	3
	32	0.970	4
	64	0.941	5
	128	0.889	4
	256	0.800	5
	512	0.667	7

K	S	Rate R	χ
16384	256	0.985	3
	512	0.970	4
	1024	0.941	5
	2048	0.889	4
	4096	0.800	6
	8192	0.667	8

Table 1 Parameters of some low-density generator matrix codes.

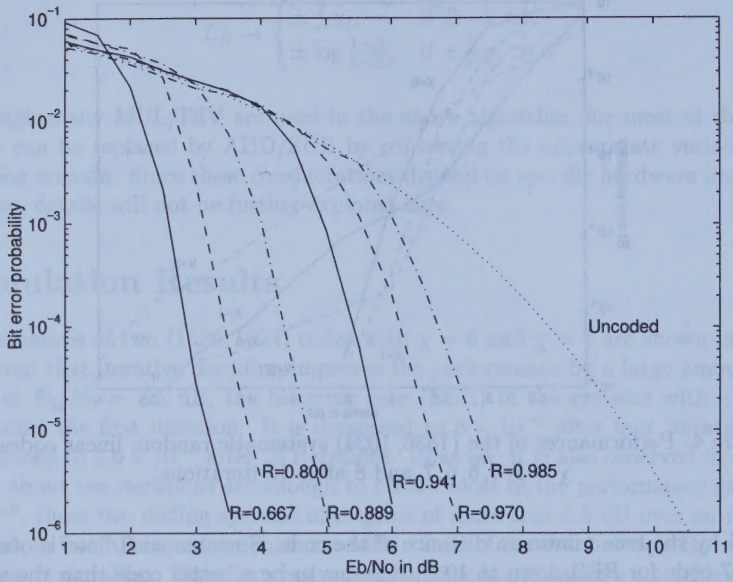


Figure 5: Performances of $(1024/R, 1024)$ systematic random linear codes after 16 iterations.

signal-to-noise ratio is low. This is exhibited in the figure that the performance curves are roughly upper-enveloped by the curve for the uncoded systems, especially for very high rate codes.

To gain a clearer view of the performances of these new codes, we have plotted in Fig. 6 the channel capacities of memoryless AWGN channels with equiprobable signaling of BPSK, QPSK, and 8PSK given by [27]

$$C^* = \log_2 M - \frac{1}{M}\sum_{i=1}^{M} E\left[\log_2\left(\sum_{j=1}^{M}\exp\frac{|s_i+n-s_j|^2-|n|^2}{N_0}\right)\right],$$

where M is the size of the signaling alphabet, s_i is a signaling symbol with $E[|s_i|^2]=1$, and n is a Gaussian random variable with zero mean and variance $N_0/2$. The performances of the codes listed in Table 1 with $K = 16384$ after 16 iterations of decoding are marked in Fig. 6 by cross-plus signs and labeled by "BPSK with LDGM Codes." Their performances are about 1 dB of the capacity as shown in the figure. The longer length thus contributes about one more dB to the coding gains above those of the codes with $K = 1024$ in Table 1. The frequency efficiencies can be easily doubled by using QPSK signaling instead of BPSK. These systems are marked in Fig. 6 by cross-plus signs and labeled by "QPSK with LDGM Codes." As shown in the figure, they achieve higher coding gains than those of Ungerboeck's trellis coded (TC) 8PSK [27] (marked by circle-plus signs) with marginally lower spectral efficiencies. The combination of QPSK and the $R = 0.941$ code achieves a coding gain higher than even the TC8PSK with a constraint length 16 code and sequential decoding [10], though with a slightly lower

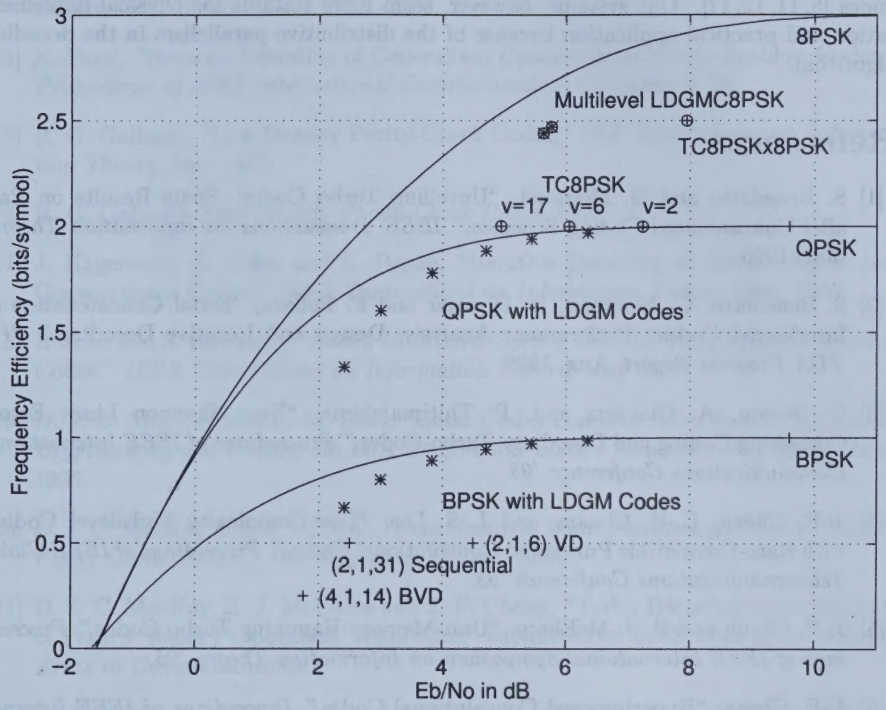


Figure 6: Spectral efficiency comparison of codes. (Marks for specific coding systems are based on the code rates and the required E_b/N_0 for BER=10^{-5}.)

spectral efficiency. It is also possible to apply these new codes to multilevel coded modulations [4,18,25] for even higher spectral efficiencies. The performances of two multilevel coded 8PSK [8] are marked in the figure by circle-cross-plus signs and labeled by "Multilevel LDGMC8PSK." Their frequency efficiencies are about 2.5 bits per symbol. They achieve coding gains that are about 1 dB of the capacity and are 2 dB higher than that of a multidimensional TC8PSK [24] (labeled by "TC8PSKx8PSK" in Fig. 6).

5 Conclusion

A class of high-rate linear binary codes based on low-density generator matrices was proposed. There codes are inherently linear-time encodable. A distributive parallel decoding algorithm of linear complexity based on the belief propagation algorithm was presented. Experimental results showed that performance approaching the channel capacities for binary and multiphase signalings. Compared with low-density parity-check codes [15,16,19,20], the proposed systems achieve similar performance but require lower encoding complexity. This is because, since H is sparse, the generator matrix G is dense and, hence, the encoding process involves $O(N^2)$ additions [19]. For lower rates,

the performances of these new codes seem to be not as good as those for the turbo codes [5, 11, 12, 17]. Our systems, however, seem more suitable for physical implementation and practical application because of the distributive parallelism in the decoding algorithm.

References

[1] S. Benedetto and G. Montorsi, "Unveiling Turbo Codes: Some Results on Parallel Concatenated Coding Schemes," *IEEE Transactions on Information Theory,* Mar. 1996.

[2] S. Benedetto, G. Montorsi, D. Divsalar and F. Pollara, "Serial Concatenation of Interleaved Codes: Performance Analysis, Design and Iterative Decoding," *JPL TDA Progress Report,* Aug. 1996.

[3] C. Berrou, A. Glavieux and, P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding and Decoding: Turbo-Codes," *Proceedings of IEEE International Communications Conference '93.*

[4] J.-F. Cheng, C.-H. Chuang and L.-S. Lee, "Low-Complexity Multilevel Coding with Rate-Compatible Punctured Convolutional Codes," *Proceedings of IEEE Global Telecommunications Conference '93.*

[5] J.-F. Cheng and R. J. McEliece, "Unit-Memory Hamming Turbo Codes," *Proceedings of IEEE International Symposium on Information Theory '95.*

[6] J.-F. Cheng, "Hyperimposed Convolutional Codes," *Proceedings of IEEE International Communications Conference '96.*

[7] J.-F. Cheng, "On the Decoding of Certain Generalized Concatenated Convolutional Codes," submitted to *IEEE Journal on Selected Areas in Communications.*

[8] J.-F. Cheng, "On the Construction of Efficient Multilevel Coded Modulations," submitted to IEEE International Symposium on Information Theory '97.

[9] F. R. K. Chung, "Constructing Random-Like Graphs," in *Probabilistic Combinatorics and Its Applications*, B. Bollobas, editor, 1991.

[10] D. J. Costello, Jr., L. C. Perez, and F. Wang, "Bandwidth Efficient CCSDS Coding Standards Proposals," *Semi-annual Status Report*, NASA Grant NAG 5-557, University of Notre Dame, Indiana, May 1992.

[11] D. Divsalar and F. Pollara, "Multiple turbo codes for deep-space communications," *JPL TDA Progress Report,* May 1995.

[12] D. Divsalar, S. Dolinar, R. J. McEliece and F. Pollara, "Transfer Function Bounds on the Performance of Turbo Codes," *JPL TDA Progress Report,* Aug. 1995.

[13] D. Divsalar and F. Pollara, "On the Design of Turbo Codes," *JPL TDA Progress Report,* Nov. 1995.

[14] K. Fazel, "Iterative Decoding of Generalized Concatenated Blokh–Zyablov–Codes," *Proceedings of IEEE International Communications Conference '96.*

[15] R. G. Gallager, "Low-Density Parity-Check Codes," *IRE Transactions on Information Theory,* Jan. 1962.

[16] R. G. Gallager, *Low-Density Parity-Check Codes,* 1963.

[17] J. Hagenauer, E. Offer and L. Papke, "Iterative Decoding of Binary Block and Convolutional Codes," *IEEE Transactions on Information Theory,* Mar. 1996.

[18] H. Imai and S. Hirakawa, "A New Multilevel Coding Method Using Error-Correcting Codes," *IEEE Transactions on Information Theory,* May 1997.

[19] D. J. C. MacKay and R. M. Neal, "Good Codes Based on Very Sparse Matrices," *Cryptography and Coding, 5th IMA Conference, Lecture Notes in Computer Science,* 1995.

[20] D. J. C. MacKay and R. M. Neal, "Near Shannon Limit Performance of Low Density Parity Check Codes," *IEE Electronic Letters,* 1996.

[21] D. J. C. MacKay, R. J. McEliece and J.-F. Cheng, "Turbo Decoding as an Instance of Peral's 'Belief Propagation' Algorithm," submitted to *IEEE Journal on Selected Areas in Communications.*

[22] R. J. McEliece, E. Rodemich and J.-F. Cheng, "The Turbo Decision Algorithm," *Proc. 33rd Allerton Conference on Communications, Control and Computing,* 1995.

[23] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference,* Morgan Kaufmann Publishers, 1988.

[24] S. S. Pietrobon, R. H. Deng, A. Lafanechere, G. Ungerboeck and D. J. Costello, Jr., "Trellis-Coded Multidimensional Phase Modulation," *IEEE Transactions on Information Theory,* Jan. 1990.

[25] S. Sayegh, "A Class of Optimum Block Codes in Signal Space," *IEEE Transactions on Communications,* Oct. 1986.

[26] R. M. Tanner, "A Recursive Approach to Low Complexity Codes," *IEEE Transactions on Information Theory,* Sep. 1981.

[27] G. Ungerboeck, "Channel Coding with Multilevel/Phase Signals," *IEEE Transactions on Information Theory,* Jan. 1982.

[28] N. Wilberg, H.-A. Loeliger and R. Kotter, "Codes and Iterative Decoding on General Graphs," *European Transactions on Telecommunications,* Sep. 1995.

Can Sequential Decoding Compete with Turbo-codes?*

Daniel J. Costello, Jr. and Hermano A. Cabral
Department of Electrical Engineering, University of Notre Dame
Notre Dame, Indiana 46556

It has long been known that nearly error-free communication over noisy channels is possible in theory but hard to achieve in practice. Claude Shannon showed in 1948[1] that if the information rate is less than the capacity of the channel, then, for any positive value ϵ, there exist codes with decoding error probability less than ϵ. Since Shannon assumed random codes in his proof, whose complexity of decoding was not defined, it soon became apparent that good codes with low enough complexity to allow a practical implementation were difficult to find, particularly for code rates close to capacity. Therefore, until recently, the usual solution to this problem was to design structured codes, making them relatively easy to implement, that yielded moderately low error probabilities at rates significantly lower than capacity. However, in 1993, a French research group[2] presented a class of codes, called *Turbo-codes*, that achieved moderately low error probabilities with reasonable decoding complexity at rates very close to capacity. Using iterative decoding, a 16-state rate 1/2 Turbo-code can operate at an SNR of 0.7dB (0.7dB greater than the capacity of the AWGN channel) with a decoded bit error rate (BER) of 10^{-5}. For comparison, the Big Viterbi Decoder (BVD), designed to decode a much more complex 16384-state code, requires an SNR of 2.4dB to achieve the same BER.

Turbo-codes are designed based on two (or more) short constraint length convolutional codes, together with one (or more) interleavers, and the decoder employs an iterative version of the MAP algorithm[3]. In order to achieve the best performance, though, one is forced to use a large interleaver size (which translates into a large block size for the code) and a large number of decoding iterations. This leads to both large delay and large computational complexity. In addition, due to the underlying short constraint length convolutional code, Turbo-codes suffer from an 'error-floor', which significantly reduces the improvement in BER with increasing SNR for BER's below 10^{-5}.

Sequential decoding of convolutional codes is one technique that allows the use of large constraint length convolutional codes, which are necessary to obtain low error probabilities. One characteristic of sequential decoders is their variable decoding time, which forces the use of a buffer to store received channel symbols for later processing, and which can cause erasures (buffer overflow). For SNR's above a limit associated with the channel cut-off rate R_0, however, the probability of erasure is typically small, and sequential decoders yield essentially error-free communication with moderate delay and computational complexity. Unfortunately, for SNR's below this limit, the computational

*This work was supported in part by NASA Grant NAG5-557, NSF Grant NCR95-22939, the Lockheed-Martin Corporation, and Brazilian Consellho Nacional de Pesquisa (CNPq) Grant 200617/94-0

complexity becomes large, preventing the use of such decoders in the SNR range between capacity and R_0. This gives Turbo-codes a performance advantage of approximately 1.7dB over sequential decoding, in the case of a code operating at a BER of 10^{-5} on an AWGN channel.

Bootstrap Hybrid Decoding[4] is a scheme that employs sequential decoders to decode a set of code sequences from a (large constraint length) convolutional code. The use of sequential decoders implies that this scheme has the properties mentioned above, i.e., a variable decoding time and the need for a buffer to store received channel symbols. However, it also has an improved cut-off rate due to the use of a parity sequence, transmitted together with the other codewords, which significantly improves the overall performance.

The Bootstrap Decoding Scheme (BDS, for short) uses $m-1$ (where m is a user-specifiable number) identical convolutional codes to encode $m-1$ binary information sequences. From these encoded binary streams, a "parity" stream, based on algebraic constraints across the encoded streams, is computed, and then all m streams are transmitted through a binary input channel. Upon receiving these streams, the decoder uses the additional information provided by the algebraic constraints to aid in the sequential decoding of the streams. Upper and lower bounds on the computational cutoff rate R_0 of the decoding scheme as a function of SNR can be derived[4][5], which show that the BDS performance, using large constraint length convolutional codes, is similar to that of turbo-codes. At rate 1/2, for example, the values for the lower and upper bounds on R_0 are 1.1dB and 0.6dB, respectively. Preliminary results of computer simulations indicate that the performance of the BDS is close to that of turbo-codes, although we have not yet demonstrated the performance promised by the bounds. Some modifications to the basic scheme are being investigated, however, in an attempt to improve the performance, e.g., the use of different rate codes, whose average rate is still 1/2, and the use of additional parity streams[5].

References

[1] C. E. Shannon, "A Mathematical Theory of Communication," *Bell Syst. Tech. J.*, 27, pp. 379–423 (Part I), 623–656 (Part II), July 1948.

[2] C. Berrou, A. Glavieux, and P. Thitimajshima, "Near Shannon Limit Error-Correcting Coding: Turbo Codes." Proc. 1993 IEEE International Conference on Communications, pp. 1064–1070.

[3] L. R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal Decoding of Linear Codes for Minimizing Symbol Error Rate," *IEEE Trans. Inform. Theory*, pp. 284–287, March 1974.

[4] F. Jelinek and J. Cocke, "Bootstrap Hybrid Decoding for Symmetrical Binary Input Channels." *Information and Control*, vol. 18, No. 3, pp. 261–298, April 1971.

[5] D. J. Costello, Jr. and F. Jelinek, "Computation Bounds on 3-group Bootstrap Decoding," *Proc.* 10^{th} *Annual Allerton Conference on Circuit and System Theory*, pp. 178–184, Oct. 1972.

Optimal Adaptive Filtering Using Cross-Validation

Robert D. Nowak *
Department of Electrical Engineering
Michigan State University
260 Engineering Building
East Lansing, MI 48824-1226 USA
(517) 355-5235
rnowak@egr.msu.edu
http://www.egr.msu.edu/~rnowak/

Abstract—**This paper develops an optimal, data-adaptive estimator of a deterministic signal in noise. The methods of penalized least squares and cross-validation balance the trade-off between bias and variance and produce a closed form expression for the estimator. The estimator is simultaneously optimal in a small-sample, predictive sum of squares sense and asymptotically optimal in the mean square sense. The framework developed in this paper enables us to assess the effect of using different signal bases in the estimation problem. We introduce the notion of a "matched basis" for a signal or class of signals. We show that estimators designed with a matched basis perform significantly better than other estimators designed with bases that represent the signal less efficiently. An application of the optimal estimator to nuclear medicine imaging is also presented.**

I. Introduction

This paper considers the problem of estimating a deterministic signal $\mathbf{s} \in \mathbb{R}^d$ in a zero mean noise $\boldsymbol{\eta}$. We assume $n > 1$, i.i.d. observations $\mathbf{x}_1, \ldots, \mathbf{x}_n$, where $\mathbf{x}_i = \mathbf{s} + \boldsymbol{\eta}_i$. The noise distribution is unknown, and we only assume $\mathrm{E}[\boldsymbol{\eta}^T \boldsymbol{\eta}] < \infty$.

The sample mean $\hat{\mathbf{s}} = n^{-1} \sum_{j=1}^{n} \mathbf{x}_j$ is an unbiased estimator of $\mathbf{s}$. We may also represent $\mathbf{s}$ with respect to the orthonormal basis $\{\mathbf{b}_i\}_{i=1}^{d}$ as $\mathbf{s} = \sum_{i=1}^{d} \theta_i \mathbf{b}_i$, where θ_i are the coordinates of the signal in this basis. In matrix notation $\mathbf{s} = \mathbf{B}\boldsymbol{\theta}$, where $\mathbf{B} = [\mathbf{b}_1, \ldots, \mathbf{b}_d]$ and $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_d]^T$. The problem of estimating $\mathbf{s}$ is then equivalent to estimating the coefficients $\boldsymbol{\theta}$.

It is well-known that estimation performance is greatly improved when the basis efficiently represents the unknown signal. Selection of an appropriate basis is one way to incorporate prior signal information into the estimation procedure. For example, smooth wavelet bases are optimal bases for estimation of signals that may contain some points of discontinuity but otherwise are smooth [4]. For the moment, we assume that the basis is fixed and concentrate on estimating $\boldsymbol{\theta}$ rather than $\mathbf{s}$ directly. In Section IV, we examine the problem of choosing a good basis. The sample estimator of $\boldsymbol{\theta}$ is

$$\hat{\boldsymbol{\theta}} = n^{-1} \sum_{j=1}^{n} \mathbf{B}^T \mathbf{x}_j \tag{1}$$

Note that $\hat{\mathbf{s}} = \mathbf{B}\hat{\boldsymbol{\theta}}$ and $\hat{\boldsymbol{\theta}} = \mathbf{B}^T \hat{\mathbf{s}}$.

*Supported by the National Science Foundation, grant no. MIP–94–57438, and the Office of Naval Research, grant no. N00014–95–1–0849.

The estimator $\hat{\boldsymbol{\theta}}$ may also be written as the solution to the least squares problem

$$\hat{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta}} \|\hat{\mathbf{s}} - \mathbf{B}\boldsymbol{\theta}\|^2, \tag{2}$$

where $\|\cdot\|$ is the Euclidean norm.

Now define the subspace $\Theta_m = \{\boldsymbol{\theta} : \theta_i = 0,\ i > m\}$ and the constrained estimator

$$\bar{\boldsymbol{\theta}} = \arg \min_{\boldsymbol{\theta} \in \Theta_m} \|\hat{\mathbf{s}} - \mathbf{B}\boldsymbol{\theta}\|^2. \tag{3}$$

The advantage of the constrained estimator is that it is insensitive to the noise component in the span of $\{\mathbf{b}_i\}_{i>m}$ and consequently may have a smaller variance than the sample estimator. However, the reduction in estimator variance may be obtained at the expense of introducing an estimator bias.

This paper considers a penalized least squares approach to balancing the trade-off between bias and variance. Using a predictive sum of squares (PRESS) criterion and the method of cross-validation [1,11], we determine the proper amount of penalization. We show that the PRESS-optimal solution can be computed in closed form and, in effect, produces an *optimal data-adaptive filter.*

The results in this paper are related to many well known methods in statistical signal processing. For example, see [1,6,11,12]. These methods are based on the linear statistical model. The linear statistical model[1] assumes a single, indirect observation of the unknown signal parameters $\boldsymbol{\theta}$:

$$\mathbf{y} = \mathbf{H}\boldsymbol{\theta} + \mathbf{n}, \tag{4}$$

where $\mathbf{H}$ is a known matrix and $\mathbf{n}$ is a vector of scalar, zero mean, i.i.d. noises. In this model, the signal $\mathbf{H}\boldsymbol{\theta}$ has a parametric form specified by the columns of $\mathbf{H}$.

Our problem formulation is quite different from the linear statistical model. We directly observe *multiple vector observations* of a deterministic signal $\mathbf{s}$, each with i.i.d. *vector noises.* The most distinguishing feature of our formulation is that *the distribution of the noise vectors is completely unknown* and the scalar noises within each vector are not necessarily i.i.d.

II. Penalized Least Squares Estimation of $\boldsymbol{\theta}$

A penalized least squares estimator for $\boldsymbol{\theta}$ is

$$\tilde{\boldsymbol{\theta}}(\boldsymbol{\Lambda}) = \arg \min_{\boldsymbol{\theta}} \|\hat{\mathbf{s}} - \mathbf{B}\boldsymbol{\theta}\|^2 + \boldsymbol{\theta}^T \boldsymbol{\Lambda} \boldsymbol{\theta}, \tag{5}$$

where $\boldsymbol{\Lambda}$ is a diagonal matrix with entries $\lambda_i \geq 0$. The term $\boldsymbol{\theta}^T \boldsymbol{\Lambda} \boldsymbol{\theta}$ is called the *penalizing functional.* Increasing λ_i increases the cost associated with signal estimates having a large component in the $\mathbf{b}_i$ direction. Hence, the penalizing functional can be designed to weight undesirable or unlikely solutions more heavily than desirable or likely solutions.

It is easily verified that the unique minimizer is $\tilde{\boldsymbol{\theta}}(\boldsymbol{\Lambda}) = (\mathbf{I} + \boldsymbol{\Lambda})^{-1} \hat{\boldsymbol{\theta}}$. Furthermore, because $\boldsymbol{\Lambda}$ is diagonal, $(\mathbf{I}+\boldsymbol{\Lambda})^{-1} = \mathbf{I}-\boldsymbol{\Gamma}$, where $\boldsymbol{\Gamma}$ is diagonal with entries $\gamma_i = \frac{\lambda_i}{1+\lambda_i}$. Since each λ_i ranges over $[0, \infty)$, we have $\gamma_i \in [0,1]$. The γ_i are referred to as *regularization parameters.*

Rewriting $\tilde{\boldsymbol{\theta}}(\boldsymbol{\Lambda})$ as a function of $\boldsymbol{\Gamma}$ produces

$$\tilde{\boldsymbol{\theta}}(\boldsymbol{\Gamma}) = \hat{\boldsymbol{\theta}} - \boldsymbol{\Gamma}\hat{\boldsymbol{\theta}}, \tag{6}$$

where $\hat{\boldsymbol{\theta}}$ is the sample mean estimator (1). Equation (6) shows that $\tilde{\boldsymbol{\theta}}(\boldsymbol{\Lambda})$ is a type of shrinkage estimator [6,12]; $\tilde{\boldsymbol{\theta}}(\boldsymbol{\Gamma})$ "shrinks" $\hat{\boldsymbol{\theta}}$ closer to the origin. Setting $\gamma_i = 0$, $i = 1, \ldots, m$ and $\gamma_i = 1$, $i = m+1, \ldots, d$ yields the constrained estimator (3).

[1]This model is also referred to as the linear regression model in the statistics literature.

Note that we can study the relationship between $\tilde{\boldsymbol{\theta}}(\boldsymbol{\Gamma})$ and $\boldsymbol{\Gamma}$ along each coordinate $\tilde{\theta}_i(\gamma_i) = \hat{\theta}_i - \gamma_i \hat{\theta}_i$ independently. The theoretically optimal γ_i, in the MSE sense, is easily expressed. The value of γ_i minimizing $\mathrm{E}[|\theta_i - \tilde{\theta}_i(\gamma_i)|^2]$ is

$$\gamma_i^{(\mathrm{MSE})} = \frac{\sigma_i^2}{\sigma_i^2 + n\,\theta_i^2}, \tag{7}$$

where $\sigma_i^2 = \mathrm{E}[(\mathbf{b}_i^T(\mathbf{x} - \mathbf{s}))^2]$ and $\theta_i^2 = (\mathbf{b}_i^T \mathbf{s})^2$, the noise and signal power, respectively, in the subspace spanned by $\mathbf{b}_i$. The MSE-optimal estimator is $\tilde{\theta}_i^{(\mathrm{MSE})} = \hat{\theta}_i - \gamma_i^{(\mathrm{MSE})} \hat{\theta}_i$. This is simply the classical Wiener filter estimate.

One approach at this point would be to compute sample estimates of σ_i^2 and θ_i^2 from the data $\mathbf{x}_1, \ldots, \mathbf{x}_n$ and then plug these estimates into (7) to obtain an estimate of $\gamma_i^{(\mathrm{MSE})}$. However, this estimate directly involves the estimate of the unknown parameter θ_i. Alternatively, we consider choosing γ_i using the method of cross-validation.

III. Estimating the Regularization Parameters by Cross-Validation

Cross-validation is a standard procedure for assessing the performance of an estimator [1]. To formulate a cross-validation procedure for this problem, first compute a penalized least squares estimate of $\boldsymbol{\theta}$ with the jth data vector $\mathbf{x}_j$ omitted: $\tilde{\boldsymbol{\theta}}(\boldsymbol{\Gamma})_j = (\mathbf{I} - \boldsymbol{\Gamma}) \frac{1}{n-1} \sum_{k \neq j} \mathbf{B}^T \mathbf{x}_k$. This "leaving-one-out" estimate is then used to predict $\mathbf{B}^T \mathbf{x}_j$. The predictive sum of squares (PRESS)

$$V(\boldsymbol{\Gamma}) = \frac{1}{n} \sum_{j=1}^{n} \|\tilde{\boldsymbol{\theta}}(\boldsymbol{\Gamma})_j - \mathbf{B}^T \mathbf{x}_j\|^2, \tag{8}$$

may be viewed as a small-sample optimality criterion measuring the quality of the estimator and of the parameters $\boldsymbol{\Gamma}$. The objective is to choose $\boldsymbol{\Gamma}$ to minimize $V(\boldsymbol{\Gamma})$.

A closed form expression for the minimizer exists. After some algebra, we have

$$V(\boldsymbol{\Gamma}) = \left(\frac{n}{n-1}\right) \sum_{i=1}^{d} \left[\hat{\sigma}_i^2 - \gamma_i\, 2n^{-1} \hat{\sigma}_i^2 + \gamma_i^2\, n^{-2} (\hat{\sigma}_i^2 + n(n-1)\hat{\theta}_i^2\right], \tag{9}$$

where $\hat{\theta}_i = n^{-1} \sum_{j=1}^{n} \mathbf{b}_i^T \mathbf{x}_j$, and unbiased estimate of θ_i, and $\hat{\sigma}_i^2 = (n-1)^{-1} \sum_{j=1}^{n} [\mathbf{b}_i^T \mathbf{x}_j - \hat{\theta}_i]^2$, an unbiased estimate of σ_i^2.

We can study the quadratic relationship between $V(\boldsymbol{\Gamma})$ and $\boldsymbol{\Gamma}$ along each coordinate independently. Setting $\frac{dV}{d\gamma_i} = 0$ and solving for γ_i produces

$$\gamma_i^{(\mathrm{min})} = \frac{\hat{\sigma}_i^2}{n^{-1}\hat{\sigma}_i^2 + (n-1)\hat{\theta}_i^2}. \tag{10}$$

Note that $\gamma_i^{(\mathrm{min})} \geq 0$. It is easily verified that $\gamma_i^{(\mathrm{min})} > 1$ implies that V is strictly decreasing on $0 \leq \gamma_i \leq 1$. Therefore, if $\gamma_i^{(\mathrm{min})} \geq 1$, then the PRESS is minimized at $\gamma_i = 1$. Hence, the PRESS-optimal choice of γ_i is

$$\gamma_i^{(\mathrm{PRESS})} = \tau\left(\frac{\hat{\sigma}_i^2}{n^{-1}\hat{\sigma}_i^2 + (n-1)\hat{\theta}_i^2}\right), \tag{11}$$

where τ is the threshold nonlinearity

$$\tau(\gamma) = \begin{cases} \gamma, & \gamma < 1 \\ 1, & \gamma \geq 1. \end{cases} \tag{12}$$

The PRESS-optimal estimator is $\widetilde{\theta}_i^{(\text{PRESS})} = \widehat{\theta}_i - \gamma_i^{(\text{PRESS})}\widehat{\theta}_i$.

Note that $\gamma_i^{(\text{PRESS})}$ is quite different from simply estimating $\gamma_i^{(\text{MSE})}$ from the sample statistics. In fact, plugging the unbiased estimates $\widehat{\sigma}_i^2$ and $\widehat{\theta}_i^2$ into (7) produces a estimate $\widehat{\gamma}_i^{(\text{MSE})} \leq \gamma_i^{(\text{PRESS})}$. Hence, the estimate $\widehat{\gamma}_i^{(\text{MSE})}$ may be over-conservative in the PRESS sense.

It is easily verified that $0 < \gamma_i^{(\text{PRESS})} < 1$ whenever $0 < n^{-1}\widehat{\sigma}_i^2 < \widehat{\theta}_i^2$. The two extremes at $\gamma_i^{(\text{PRESS})} = 1$ and $\gamma_i^{(\text{PRESS})} = 0$ are especially interesting.

$\gamma_i^{(\text{PRESS})} = 1$ ($\widetilde{\theta}_i^{(\text{PRESS})} = 0$) if and only if $\widehat{\theta}_i^2 \leq n^{-1}\widehat{\sigma}_i^2$. Given n i.i.d. observations, the quantity $n^{-1}\widehat{\sigma}_i^2$ is an estimate of the averaged noise power in the subspace $\mathcal{S}_i$ spanned by $\mathbf{b}_i$ (the true noise power is $n^{-1}\sigma_i^2$). $\widehat{\theta}_i^2$ is an estimate signal power in $\mathcal{S}_i$. Hence, the PRESS-optimal estimate of θ_i is zero if and only if the SNR in $\mathcal{S}_i$ is less than or equal to 0dB.

$\gamma_i^{(\text{PRESS})} = 0$ ($\widetilde{\theta}_i^{(\text{PRESS})} = \widehat{\theta}_i$) if and only if $\widehat{\sigma}_i^2 = 0$. This shows that the sample mean estimator $\widehat{\theta}_i$ is suboptimal in the PRESS sense *except* when there is no noise in $\mathcal{S}_i$. This situation does not occur very often: In [9] it is shown that if the distribution of $\mathbf{x}$ is absolutely continuous, then $\text{Prob}[\gamma_i^{(\text{PRESS})} = 0] = 0$. It is also shown that under the same condition $\text{Prob}[\gamma_i^{(\text{MSE})} = 0] = 0$, where $\gamma_i^{(\text{MSE})}$ is the MSE-optimal regularization parameter (7). One interpretation of this result is the following. If $\mathbf{x}$ is absolutely continuous, then it is always possible to reduce the estimator variance by an amount $\epsilon > 0$ at the expense of adding a squared bias $< \epsilon$. This type of result is widely known in the statistical literature [6].

In addition to the PRESS-optimality of $\widetilde{\boldsymbol{\theta}}^{(\text{PRESS})}$, we can show that $\widetilde{\boldsymbol{\theta}}^{(\text{PRESS})}$ is asymptotically optimal in the mean square sense, that is $\gamma_i^{(\text{PRESS})} / \gamma_i^{(\text{MSE})} \rightarrow 1$ *w.p.* 1. The result is obtained by a simple application of the strong law of large numbers; the details are given in [9]. Arguing along similar lines, it is easily established that $\widetilde{\boldsymbol{\theta}}^{(\text{PRESS})} \rightarrow \mathbf{B}^T\mathbf{s}$ *w.p.* 1.

IV. Choosing a Basis

Note that the PRESS and MSE optimality of the PRESS-optimal estimator is *with respect to the basis* $\mathbf{B}$. Although the rate of convergence does not depend on $\mathbf{B}$, the small-sample error is affected by the choice $\mathbf{B}$. Intuition suggests that the PRESS-optimal estimator will provide the best performance (small PRESS and MSE) if the basis $\mathbf{B}$ is well-matched to the unknown signal. Although it is difficult to characterize the PRESS-optimal basis, this intuition is partially justified by considering the MSE-optimal basis. Recall, the PRESS-optimal estimator performs in a similar manner as the number of observations increases.

MSE-Optimal Basis

Consider the MSE as a function of the underlying signal basis $\mathbf{B}$. The MSE of the MSE-optimal estimator (7) can be written as

$$\text{MSE}(\mathbf{B}) = \text{E}\left[\|\mathbf{B}\widetilde{\boldsymbol{\theta}}^{(\text{MSE})} - \mathbf{s}\|_2^2\right] = \sum_{i=1}^{d} \text{E}\left[|\theta_i - \widetilde{\theta}_i^{(\text{MSE})}|^2\right], \tag{13}$$

where we have used the fact that $\mathbf{B}$ is an orthonormal basis. Substituting $\widetilde{\theta}_i^{(\text{MSE})} = \widehat{\theta}_i - \gamma_i^{(\text{MSE})}\widehat{\theta}_i$, we get

$$\text{E}\left[|\theta_i - \widetilde{\theta}_i^{(\text{MSE})}|^2\right] = n^{-1}\sigma_i^2 - \frac{n^{-1}\sigma_i^4}{\sigma_i^2 + n\theta_i^2}, \tag{14}$$

where $\sigma_i^2 = \mathrm{E}[\mathbf{b}_i^T \mathbf{C} \mathbf{b}_i]$ and $\mathbf{C} = \mathrm{E}[\boldsymbol{\eta}\boldsymbol{\eta}^T]$, the covariance of the vector noise $\boldsymbol{\eta}$. Hence, the MSE of the estimator (7) is

$$\mathrm{MSE}(\mathbf{B}) = n^{-1} \operatorname{tr}[\mathbf{B}^T \mathbf{C} \mathbf{B}] - \sum_{i=1}^{d} \frac{n^{-1}\sigma_i^4}{\sigma_i^2 + n\theta_i^2}, \tag{15}$$

where tr denotes the trace operator. Note that since $\mathbf{B}$ is orthogonal $\operatorname{tr}[\mathbf{B}^T \mathbf{C} \mathbf{B}] = \operatorname{tr}[\mathbf{C}]$, independent of the basis. The quantity $n^{-1} \operatorname{tr}[\mathbf{C}]$ is the MSE of the sample estimator $\widehat{\boldsymbol{\theta}}$. Clearly then, the minimum MSE is achieved by a basis that maximizes

$$Q(\mathbf{B}) = \sum_{i=1}^{d} \frac{n^{-1}\sigma_i^4}{\sigma_i^2 + n\theta_i^2}. \tag{16}$$

$Q(\mathbf{B})$ is non-negative. Therefore, the MSE of $\widetilde{\boldsymbol{\theta}}^{(\mathrm{MSE})}$, the MSE-optimal estimator (7), is always less than or equal to the MSE of the sample estimator. $Q(\mathbf{B})$ may be viewed as a measure of the quality of a signal basis.

Maximizing Q over all orthonormal bases is a formidable task. In general, the optimal basis depends on $\mathbf{C}$, n, and the signal $\mathbf{s}$. Also notice that $Q(\mathbf{B})$ is $\mathcal{O}(n^{-2})$, as $n \to \infty$. The MSE is $\mathcal{O}(n^{-1})$. Therefore, *the choice of basis does not affect the rate of convergence.*[2] The choice of basis does, however, affect the small sample error.

For example, consider a basis $\mathbf{B}$ with the one basis vector in the direction of the signal, $\mathbf{b}_1 = \mathbf{s}/\|\mathbf{s}\|$. We call such a basis an *exactly matched basis*, since one of the basis elements is perfectly aligned with the signal. Note that there is not a unique exactly matched basis since only one element is uniquely specified and the others may be any other orthogonal set. With an exactly matched basis

$$Q(\mathbf{B}) = \frac{n^{-1}\sigma_1^4}{\sigma_1^2 + n\|\mathbf{s}\|^2} + \sum_{i=2}^{d} n^{-1}\sigma_i^2, \tag{17}$$

and

$$\mathrm{MSE}(\mathbf{B}) = n^{-1}\sigma_1^2 - \frac{n^{-1}\sigma_1^4}{\sigma_1^2 + n\|\mathbf{s}\|^2}. \tag{18}$$

In this case, the MSE is reduced by $\sum_{i=2}^{d} n^{-1}\sigma_i^2 + \frac{n^{-1}\sigma_1^4}{\sigma_1^2 + n\|\mathbf{s}\|^2}$, compared to the sample estimator MSE. The sum $\sum_{i=2}^{d} n^{-1}\sigma_i^2$ the total noise power outside the signal subspace. This shows that an exactly matched basis can drastically reduce the MSE.

However, in general, an exactly matched basis is not optimal as the following counter-example shows. Consider the simple two-dimensional case with a signal $\mathbf{s} = \left[\frac{1}{\sqrt{10}} \;\; 0\right]^T$, $n = 10$ observations, and noise covariance

$$\mathbf{C} = \begin{bmatrix} 1.0 & 0.9 \\ 0.9 & 1.0 \end{bmatrix}. \tag{19}$$

Using the exactly matched basis

$$\mathbf{B} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

we get $\mathrm{MSE}(\mathbf{B}) = 0.050$. With the basis

$$\mathbf{B} = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix},$$

[2] Keep in mind that we are considering a *parametric* estimation problem. Our signal space is finite dimensional. In *non-parametric* estimation, the choice of basis can profoundly affect the rate of convergence [4].

which is not exactly matched, we get MSE($\mathbf{B}$) = 0.048. The difference in MSE's is slight and in general one expects that a basis that is able to represent the signal with a relatively small number of basis elements will provide very good results.

To further investigate the choice of basis issue we consider the special case in which the noise is white.

WHITE NOISE

Suppose that the noise is white; that is, $\mathbf{C} = \sigma^2\mathbf{I}$, where $\mathbf{I}$ is the identity. Then $\sigma_i^2 = \sigma^2$, $\forall i$, independent of the basis. Consider an exactly matched basis $\mathbf{B}$ with the one basis vector in the direction of the signal, $\mathbf{b}_1 = \mathbf{s}/\|\mathbf{s}\|$. Then

$$Q(\mathbf{B}) = \frac{n^{-1}\sigma^4}{\sigma^2 + n\|\mathbf{s}\|^2} + \sum_{i=2}^{d} n^{-1}\sigma^2, \tag{20}$$

and

$$\mathrm{MSE}(\mathbf{B}) = n^{-1}\sigma^2 - \frac{n^{-1}\sigma^4}{\sigma^2 + n\|\mathbf{s}\|^2}. \tag{21}$$

In this case, the MSE is reduced by a factor greater than d, the dimension of the signal. If d is large, then the reduction can be quite significant. This is the best that we can do in this case as the following proposition, proved in [9], shows.

Proposition: If the noise is white, then MSE($\mathbf{B}$) is minimized by every orthonormal basis with one basis vector in the direction of the signal.

For comparison, consider the other extreme in which the signal energy is evenly distributed over the coefficients. That is, $\theta_i^2 = \|\mathbf{s}\|^2/d$. The SNR in each coefficient is also constant, equal to $\frac{\|\mathbf{s}\|^2}{n^{-1}d\,\sigma^2}$. If the SNR is 0dB, then equation (15) shows that the MSE is reduced by a factor of 1/2, compared to the sample estimator MSE. Hence, even when the basis does not align with the signal the MSE-optimal estimator performs better than the sample estimator.

MATCHED BASES

The exactly matched basis requires perfect knowledge of the signal direction. In practice, such knowledge is usually not available. We may have partial prior knowledge that suggests that a particular basis may be a better match to the signal than other bases. Specifically, we define the notion of a *matched basis.*

Definition: A basis $\mathbf{B}$ is called a *matched basis* for a signal $\mathbf{s}$ if most of the signal energy is concentrated in a relatively small number of coefficients in that basis.

The definition of a matched basis is intentionally non-precise because the prior information that is typically available in practice is non-precise. A matched basis does not mean that the signal must lie completely in the span of a subset of basis vectors. In general, the better matched the basis is to the signal, the more the MSE is reduced.

To study this effect, we computed MSE($\mathbf{B}$) for several bases at various SNRs. We again assume the noise is white and define SNR = $\frac{\|\mathbf{s}\|^2}{n^{-1}d\,\sigma^2}$. We consider bases that match the signal to varying degrees. To accomplish this, we start with the signal energy uniformly distributed over the coefficients. We then skew this energy distribution by reducing the energy in a certain percentage of the coefficients by an amount δ^2 and add the total amount removed from these coefficients, uniformly, to the remaining coefficients. For example, we begin with $|\theta_i|^2 = \|\mathbf{s}\|^2/d$, $\forall i$. Then subtract $q\,\|\mathbf{s}\|^2/d$, $0 < q < 1$, from $p\,d$, $0 < p < 1$, coefficients and add $\frac{qp}{1-p}\,\|\mathbf{s}\|^2/d$ to the remaining $(1-p)\,d$ coefficients. The larger p and q are, the more closely matched the basis is to the signal. With $q = 1$ and $p = \frac{d-1}{d}$, we have an exactly matched basis with one of the basis elements in the direction of the signal.

Figure 1 depicts the relationship between the MSE and the basis. The degree to which the basis is matched to the signal is evaluated with $p = 0.00, 0.75, 0.95, 0.99$ and $q = 0.99$. Each pair (p, q) generates a curve of the MSE as a function of the SNR. The total signal power is unity in all cases. The noise power is adjusted to produce the various SNRs. The uppermost curve (a straight line) is the sample estimator MSE. The five lower curves correspond to the MSE-optimal estimators, each with a different basis. As p increases, the MSE decreases. The lowermost curve is the the MSE resulting from an exactly matched basis. This is a lower bound on the performance of the MSE-optimal estimator and depends on the dimensionality of the problem (in this case $d = 100$). As the dimension increases, this lower bound decreases. Except in the exactly matched case, the MSE-optimal estimators are *not* equivalent to a truncated series estimator. All coefficients contain some fraction of the total signal energy. The MSE-optimal estimator "shrinks" each coefficient value in proportion to SNR in that coordinate, but does not set small coefficients to zero. In general, *the MSE-optimal estimate does not lie entirely in the span of a subset of the basis vectors.*

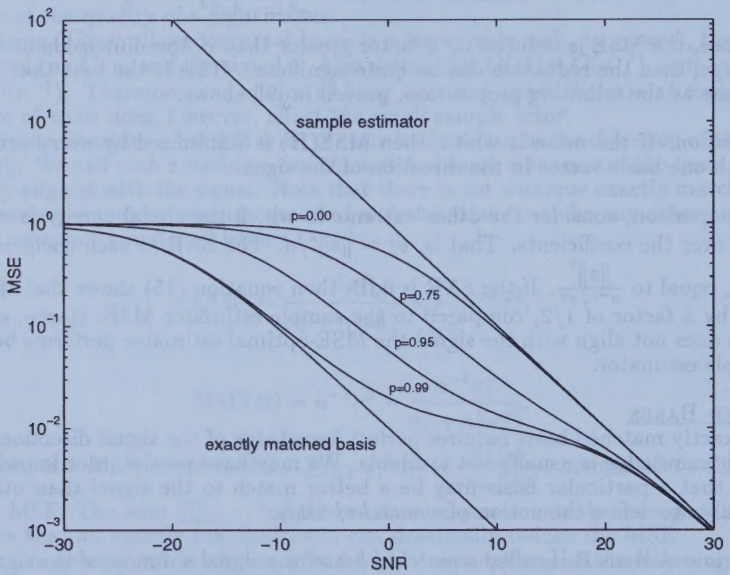


Figure 1: *Effect of matched basis on MSE. Larger values of p correspond MSE-optimal estimators using a basis that is more closely matched to the signal.*

From Figure 1, we see that the better matched the basis is to the signal, the more the MSE is reduced. At very low SNR, the signal power is so weak that the MSE-optimal estimator shrinks the estimate to zero — annihilating both signal and noise. At this extreme, the MSE is due entirely to a squared bias equal to the signal energy. Note that the sample estimator does not adjust to the SNR and its MSE is unbounded as the SNR decreases. On the other hand, at high SNR, the signal dominates the noise and all estimators, including the sample estimator, perform very well. The exact matched basis estimator performs better than the other estimators since, even at high SNR, it completely rejects *all* noise outside the one dimensional signal subspace.

The interesting behavior occurs at more reasonable SNRs. For example, at $-10\text{db} \leq \text{SNR} \leq 10\text{db}$ we see the dramatic improvement in the MSE of the MSE-optimal estimator compared to the sample estimator. In some cases, the improvement is greater than an order of magnitude. The MSE-optimal estimator balances the trade-off between bias and variance to reduce the MSE. We infer from Figure 1 that *we do very well with a*

matched basis that is not exactly aligned with the signal. Hence, imprecise prior knowledge of the signal structure can be used to design or choose a very effective matched basis.

Examples of Matched Bases

There are many familiar examples of matched bases. For example, if the signal is approximately bandpass, then the Fourier basis is a matched basis. If the signal is sufficiently smooth, then a variety of bases will be matched to the signal, including the Fourier basis and smooth wavelet bases.

Wavelet bases are exceptional in the sense that they are matched bases for large classes of functions, over many types and degrees of smoothness, simultaneously. This is not the case for the Fourier basis [4]. The intuitive reason for the outstanding performance of wavelet bases is that they satisfy smoothness (moment) conditions and have local support. Therefore, smooth wavelet bases are a natural representation of signals that are generally smooth except for some number of singularities. This heuristic is attributed to Mallat [5]. The theoretical justification behind this intuition is studied in [5].

In general, specific problems may dictate a specific form for the matched basis. The task of choosing a matched basis is also closely related to the "best basis" problem [2]. If we have partial knowledge of the class of signals involved, then it may be possible to choose a good matched basis using this information.

V. Application to Nuclear Medicine Imaging

We propose an new method for image estimation in static nuclear medicine studies. The new method is based on splitting the observation period T into n disjoint time intervals, each of duration T/n. The counts detected in each time interval produce an independent image of the underlying Poisson process. Hence, we have n independent images of the underlying process, each with roughly $1/n$ of the total counts detected over the entire observation period T. Let $\mathbf{x}_i$, $i = 1, \ldots, n$, denote these n images. Each image has the form

$$\mathbf{x}_i = \mathbf{s} + \boldsymbol{\eta}_i, \tag{22}$$

where $\mathbf{s}$ is the intensity of the Poisson process and $\boldsymbol{\eta}_i$ is a noise due to the finite number of counts, scattering, attenuation, and possible electronic noises in the detector. The dimension of the observation vector is d, the number of detectors.

The noise is non-white and, moreover, is directly correlated with $\mathbf{s}$. One approach to remedy this difficultly is to work with the square root of the counts at each detector. Asymptotically, this "variance stabilizing transformation" whitens the Poisson noise so that elements of $\mathbf{x}$ have equal variances. However, simulated and empirical studies have shown that the limited number of counts typically encountered in nuclear medicine are not sufficient to reasonably approximate this asymptotic condition [10]. Therefore, in nuclear medicine imaging we are faced with a colored noise situation.

Using a simple Poisson model for the noise, the Maximum Likelihood Estimate (MLE) of the process is the average of these n independent images, the sample estimator

$$\hat{\mathbf{s}} = \frac{1}{n} \sum_{i=1}^{n} \mathbf{x}_i. \tag{23}$$

Note that $n\hat{\mathbf{s}}$ is simply the image of the total counts detected over the entire observation period T, the standard image used in practice.

Because of the relatively small number of counts in typical nuclear medicine studies, the MLE image appears noisy and in fact has a large variance about its true mean (the underlying intensity of the Poisson process). The variance is proportional to the signal itself and hence linear filtering of the MLE may result in unsatisfactory signal degradation. The MLE image can be improved using the PRESS-optimal estimator. In effect, the PRESS-optimal estimator is an optimal, data-adaptive filter. For this application, we have chosen the Daubechies D6 wavelet basis for the PRESS-optimal estimator. This basis is selected for its good localization and smoothness properties [3]. Empirical studies have also shown that the D6 wavelet basis is a matched basis for a wide variety of nuclear medicine images [10].

In Figure 2, we show the results in one case.[3] The data for this example was collected using a "dynamic acquisition." Such acquisitions are usually used to collect multiple images of time-varying nuclear medicine studies. For our purposes, the dynamic acquistion is used to collect $n = 10$, i.i.d. images of a static nuclear medicine study. We are then able to compute the PRESS-optimal estimate from these independent images. We can see from Figure 2 that visual quality of the PRESS-optimal estimator is much better than that of the MLE.

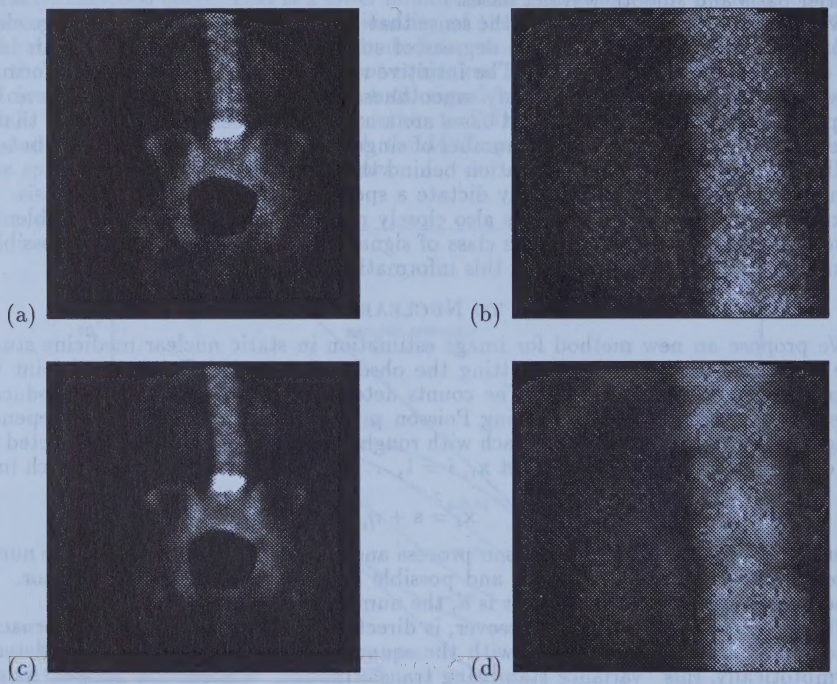


Figure 2: *Nuclear medicine imaging using the PRESS-optimal estimator. In practice, the Maximum Likelihood Estimate (MLE) is used for diagnosis. The noise in nuclear medicine is highly non-Gaussian and, moreover, is highly correlated with the signal itself rendering the noise removal task extremely difficult. The PRESS-optimal estimator has significant theoretical advantages over the MLE and generally produces much better estimates of the underlying human anatomy and physiology. (a) MLE of spine study. (b) Close-up MLE of spine study. (c) PRESS-optimal estimate of spine study. (d) Close-up PRESS-optimal estimate of spine study.*

VI. Related Work and Conclusions

The PRESS-optimal estimator is closely related to James-Stein and other "shrinkage" estimators [4,6–8,11,12]. The idea of shrinking an estimator towards the origin in order to improve the MSE was first proposed in [8]. The PRESS-optimal estimator is a special form of shrinkage estimator. Stone was apparently the first to draw the connection between shrinkage estimators and cross-validation [11]. Shrinkage estimators typically

[3]The author wishes to thank Prof. Robert S. Hellman, M.D., Division of Nuclear Medicine, Medical College of Wisconsin, for providing this data.

assume i.i.d. noises. A distinguishing feature of PRESS-optimal estimator is that *the distribution of the noise vectors is completely unknown* and the scalar noises within each vector are not necessarily i.i.d.

Another interesting feature of the PRESS-optimal estimator is that the amount of shrinkage is determined independently for each coordinate and this amount is related to the estimated SNR in the coordinate. In this respect, the PRESS-optimal estimator is similar to the data adaptive rank shaping methods developed in [12]. The methods in [12] are based on the linear statistical model and assume an additive white noise with known variance. In contrast, the PRESS-optimal estimator does not require any prior knowledge of the noise, except that the noise is zero mean has finite second order moments. We do, however, require $n > 1$ i.i.d. noisy observations of the signal. The methods of [12] only require a single observation. Also, the PRESS-optimal estimator is simultaneously optimal in the small-sample PRESS sense and asymptotically optimal in the MSE sense.

References

[1] D. M. Allen, "The relationship between variable selection and data agumentation and a method for prediction," *Technometrics*, vol. 16, no. 1, Feb. 1974.

[2] R. R. Coifman and M. V. Wickerhauser, "Entropy-based algorithms for best basis selection," *IEEE Trans. on Info. Theory*, vol. 38, no. 2, March 1992.

[3] I. Daubechies, *Ten Lectures on Wavelets*, CBMS-NSF Regional Conference Series in Applied Mathematics, no. 61, SIAM, 1992.

[4] D. L. Donoho and I.M. Johnstone, "Adapting to unknown smoothness via wavelet shrinkage," *J. American Stat. Assoc.*, pp. 1200-1224, vol. 90, no. 432, December 1995.

[5] D. L. Donoho, "Unconditional bases are optimal bases for data compression and for statistical estimation," *Applied and Computational Harmonic Analysis*, pp. 100-115, vol. 1, no. 1, December 1993.

[6] N. R. Draper and R. C. Van Nostrand, "Ridge regression and James-Stein estimation: review and comments," *Technometrics*, vol. 21, no. 4, Nov. 1979.

[7] A. E. Hoerl and R. W. Kennard, "Ridge regression: biased estimation for nonorthogonal problems," *Technometrics*, vol. 12, no. 1, pp. 55-67, Feb. 1970.

[8] W. James and C. Stein, "Estimation with quadratic loss," *Proc. Fourth Berkeley Symp. Math. Statist. Probab.*, vol. 1, pp. 473-484, 1961.

[9] R. D. Nowak, "Optimal signal estimation using cross-validation," *Rice University Technical Report*, ECE-TR-9601, Rice University, Houston, TX, Jan. 1996.

[10] R. D. Nowak, R. S. Hellman, D. J. Nowak, and R. G. Baraniuk, "Optimal estimation of nuclear medicine images using cross-validation," to appear at *IEEE Nuclear Science and Medical Imaging Symposium*, November 1996.

[11] M. Stone, Discussion, *J. Roy. Statist. Soc., B-35*, pp. 408-409, 1973.

[12] A. J. Thorpe and L. L. Scharf, "Data adaptive rank-shaping methods for solving least squares problems," *IEEE Tran. Signal Proc.*, vol. 43, no. 7, pp. 1591-1601, July 1995.

A Markov Analysis of the Second-Order Digital Phase Locked Loop

Michael P. Fitz*(fitz@ee.eng.ohio-state.edu)
Department of Electrical Engineering, Ohio State University, OH 43210

Ramakrishnan Balasubramanian†(rbalasub@ecn.purdue.edu)
James V. Krogmeier‡(jvk@ecn.purdue.edu)
School of Electrical Engineering, Purdue University, IN 47907

Abstract

This paper presents a rigorous, non-linear (Markov) analysis of a digital phase locked loop (DPLL) including quantization effects. A non-standard formulation of the problem reduces the complexity of the Chapman-Kolmogorov equation to a level where the analysis is feasible. A statistical characterization of the standard loop is presented to demonstrate the theory.

1 Introduction

Carrier synchronization or phase estimation synchronization of communication systems is an important requirement for achieving optimal demodulator performance. Communication systems frequently use DPLLs [1] to achieve carrier synchronization. The presence of a frequency offset produces a loop stress on the first-order loop and the steady state phase error is not zero, even in the absence of noise. This phase offset can produce significant degradation in system performance. Hence, the second-order DPLL is commonly used in communication systems, since it can accommodate frequency offsets without a steady state error.

This work presents the first completely rigorous analysis of the second order PLL. The analysis in [2, 3] for the analog PLLs produces only approximate characterization due to the complexity of the vector diffusion approximation formulation. Another approximate analysis of the non-uniformly sampled second order DPLL has been performed (without considering quantization effects) in [4]. It is well known that the second order PLL is characterized by a vector Markov process where the two components are the phase estimate and the frequency estimate (i.e., the DPLL's state variables). For DPLLs with reasonable quantization levels the probability mass function (pmf) at each iteration and

*Supported by National Science Foundation under Grant 9406073-NCR.
†Supported by National Science Foundation under Grant 9115820-NCR
‡Supported by an NEC faculty fellowship.

the state transition matrix can be quite large for this two dimensional process. This large size has generally precluded attempts at characterizing performance with analysis. This paper considers a very general model of the DPLL (including quantization) and identifies a technique to significantly reduce the complexity of the analysis (typically two orders of magnitude).

2 Signal and Loop Models

The analytical baseband model of the DPLL is given in Fig. 1. The input signal to the DPLL has an unknown phase and frequency offset and is given as

$$\begin{aligned} x_k &= e^{\jmath(\Omega k+\theta_0)} + n_k, \\ &= e^{\jmath\theta_k} + n_k, \end{aligned} \tag{1}$$

where n_k is an i.i.d zero mean complex Gaussian noise with variance $\frac{N_0}{E_s}$ (E_s is the energy per symbol), Ω is the normalized carrier frequency offset, θ_0 is the carrier phase at $k = 0$, and θ_k is the total carrier phase at time k.

An important variable in the DPLL performance analysis is the derotated input, v_k, which has the form,

$$\begin{aligned} v_k &= e^{\jmath(\theta_k-\hat{\theta}_k)} + \gamma_k, \\ &= e^{\jmath\phi_k} + \gamma_k, \end{aligned} \tag{2}$$

where $\hat{\theta}_k$ is the phase estimate at time k, γ_k is the rotated noise vector and ϕ_k is the phase error at time k. It should be noted that n_k and γ_k have the same statistical characterization due to the rotational invariance of complex Gaussian noise.

The DPLL's performance is determined by the phase detector (PD) and the loop filter. This paper considers the sawtooth PD given by

$$g(x_k, \hat{\theta}_k) = \varphi_k = \arg\left\{x_k e^{-\jmath\hat{\theta}_k}\right\} = \arg\{v_k\}. \tag{3}$$

The loop filter is assumed to have an "integral plus proportional form", i.e.,

$$F(z,k) = \mu_1 + \mu_2 \frac{1}{z-1}. \tag{4}$$

The above equation results in the following input-output open loop relation

$$\hat{\theta}_{k+1} - 2\hat{\theta}_k + \hat{\theta}_{k-1} = \mu_1\varphi_k + (\mu_2 - \mu_1)\varphi_{k-1}, \tag{5}$$

and the closed loop non-linear difference equation given as

$$\hat{\theta}_{k+1} - 2\hat{\theta}_k + \hat{\theta}_{k-1} = \mu_1 g\left(x_k, \hat{\theta}_k\right) + (\mu_2 - \mu_1) g\left(x_{k-1}, \hat{\theta}_{k-1}\right). \tag{6}$$

A useful reparametrization is

$$\begin{bmatrix} \hat{\theta}_{k+1} \\ \hat{\Omega}_{k+1} \end{bmatrix} = \begin{bmatrix} \hat{\theta}_k + \mu_1 g\left(x_k, \hat{\theta}_k\right) + \hat{\Omega}_k \\ \hat{\Omega}_k + \mu_2 g\left(x_k, \hat{\theta}_k\right) \end{bmatrix} \tag{7}$$

where $\hat{\Omega}_k$ (the output from the "integral" portion of the loop filter) can be thought of as the recursive estimate of Ω at time k. Note all the additions of the phase angle made in this paper are made modulo 2π.

A linear analysis of the constant gain DPLL provides some intuition into the behavior of the loop. The DPLL is linearized by ignoring noise and linearizing $g\left(x_k, \hat{\theta}_k\right)$ at $\theta_k - \hat{\theta}_k = 0$ in (6), producing a second order linear difference equation. The linear stability region of the constant gain DPLL (the values of μ_1 and μ_2 that produce poles inside the unit circle) is defined by,

$$\mu_2 < \mu_1, \quad \mu_2 \geq 0, \quad \mu_2 > 2\mu_1 - 4. \tag{8}$$

The DPLL is under, critical or over damped depending on whether $\mu_1^2 <, =, > 4\mu_2$ respectively. The two-sided loop noise bandwidth is given as

$$B_N = \frac{2\mu_1^2 + 2\mu_2^2 - 3\mu_1\mu_2 + 3\mu_2^2}{(\mu_1 - \mu_2)(4 - 2\mu_1 + \mu_2)}. \tag{9}$$

The reciprocal of the B_N provides a good approximation to the SNR improvement provided by the DPLL.

3 Markov Analysis

In order to perform a Markov analysis of a second order DPLL the non-linear effects of quantization must be considered. To this end, a more accurate model of the loop is given in Fig. 2. The function $g\left(x_k, \hat{\theta}_k\right)$ is the non-linear function representing the de-rotation and phase detection with quantization, while $f_1(\mu_1, \varphi_k)$ and $f_2(\mu_2, \varphi_k)$ represent the nonlinear (due to quantization) gain terms in the loop filter. Since quantization is integral to the analysis, we assume uniform quantization and define N_1, N_2 and N_3 as the number of quantization levels for $\hat{\theta}_k$, $\hat{\Omega}_k$ and φ_k respectively. The probability space of $\hat{\theta}_k$, $\hat{\Omega}_k$ and φ_k are denoted by Θ, Υ and Ψ respectively. The vector difference equation describing the state variables characterizing the DPLL is then

$$\hat{\Phi}_{k+1} = \begin{bmatrix} \hat{\theta}_{k+1} \\ \hat{\Omega}_{k+1} \end{bmatrix} = \begin{bmatrix} \hat{\theta}_k + f_1(\mu_1, \varphi_k) + \hat{\Omega}_k \\ \hat{\Omega}_k + f_2(\mu_2, \varphi_k) \end{bmatrix} = \mathbf{H}(\hat{\Phi}_k, x_k) \tag{10}$$

Since n_k is an i.i.d random variable, the vector process $\hat{\Phi}_k$ is a Markov random chain. This chain can be completely described by [5, 6] by the initial probability mass function (pmf), $p(\hat{\Phi}_0)$, and the Chapman-Kolmogorov equation given as

$$p(\hat{\Phi}_{k+1}) = \mathbf{S}_k \times p(\hat{\Phi}_k), \tag{11}$$

where $\mathbf{S}_k$ is the state transition matrix.

Calculating the entries in $\mathbf{S}_k$ and using traditional Markov theory can prove to be a numerically daunting task. Since $\hat{\Phi}_k$ is a $N_1 \cdot N_2$ sized vector, the state transition matrix of this vector has $N_1^2 \cdot N_2^2$ entries. Numerical calculations like those presented in [4, 7] would not be feasible for reasonable quantization levels. Fortunately due to the form of

a DPLL, the time update of the state vector pmf can be implemented in a more efficient manner than given by (11). We will use the following notation to define the entries in $\mathbf{S}_k$,

$$P_s\left(\hat{\Phi}_{k+1} \mid \hat{\Phi}_k, \theta_k\right) = P_s\left(\hat{\theta}_{k+1}, \hat{\Omega}_{k+1} \mid \hat{\theta}_k, \hat{\Omega}_k, \theta_k\right), \qquad \hat{\Phi}_k \in \Theta \times \Upsilon \tag{12}$$

Since the performance of the DPLL is not dependent on the value of θ_k and to simplify notation, the rest of the paper does not explicitly condition on θ_k. The simplification in computing the values in (12) is shown by first using total probability over φ_k on the entries of $\mathbf{S}_k$ to get

$$P_s\left(\hat{\theta}_{k+1}, \hat{\Omega}_{k+1} \mid \hat{\theta}_k, \hat{\Omega}_k\right) = \sum_{\varphi_k \in \Psi} P_s\left(\hat{\theta}_{k+1}, \hat{\Omega}_{k+1} \mid \hat{\theta}_k, \hat{\Omega}_k, \varphi_k\right) p\left(\varphi_k \mid \hat{\theta}_k, \hat{\Omega}_k\right). \tag{13}$$

It should be noted that φ_k is not a function of $\hat{\Omega}_k$ and that $\hat{\Phi}_{k+1}$ is completely deterministic given $\hat{\Phi}_k$ and φ_k. This simplifies (13) to

$$P_s\left(\hat{\theta}_{k+1}, \hat{\Omega}_{k+1} \mid \hat{\theta}_k, \hat{\Omega}_k\right) = \sum_{\varphi_k \in \Psi} 1_{\left\{\hat{\theta}_k + \mu_1(\varphi_k) + \hat{\Omega}_k; \hat{\Omega}_k + \mu_2(\varphi_k)\right\}}\left(\hat{\theta}_{k+1}, \hat{\Omega}_{k+1}\right) p\left(\varphi_k \mid \hat{\theta}_k\right), \tag{14}$$

where $1_A(\theta)$ is the indicator function. It can be seen from (14) that the entries in $\mathbf{S}_k$ are completely determined by $p\left(\varphi_k \mid \hat{\theta}_k\right)$.The pmf $p\left(\varphi_k \mid \hat{\theta}_k\right)$ can take on $N_3 \cdot N_1$ values.

The form of the state transition matrix in (14) permits an efficient update of the state vector pmf. Fig. 3 shows the pseudo code for this algorithm. The important thing about this algorithm is that the computation of the state vector pmf update is reduced from $O\left(N_1^2 \cdot N_2^2\right)$ operations to $O\left(N_1 \cdot N_2 \cdot N_3\right)$ operations. This gives at least two orders of magnitude reduction in computation, for normal levels of quantization.

Finally we need an expression for $p\left(\varphi_k \mid \hat{\theta}_k\right)$ to complete the Markov analysis. The saw tooth PD considered in this paper has the functional form

$$\varphi_k = \arg\left(e^{\jmath\phi_k} + \gamma_k\right). \tag{15}$$

The saw tooth PD produces $\Psi \subseteq (-\pi, \pi]$, and since φ_k is the phase angle of a sinewave in noise the probability density function (assuming infinite quantization [8]) is

$$p\left(\varphi_k \mid \hat{\theta}_k\right) = \frac{\exp\left(-\frac{E_s}{N_0}\right)}{2\pi} + \sqrt{\frac{E_s}{4\pi N_0}} \cos\left(\varphi_k - \phi_k\right) \times \exp\left[-\frac{E_s}{N_0} \sin^2\left(\varphi_k - \phi_k\right)\right] \cdot \left[1 + \operatorname{erf}\left(\sqrt{\frac{E_s}{N_0}} \cos\left(\varphi_k - \phi_k\right)\right)\right]. \tag{16}$$

It should be noted that the actual pmf of φ_k used for this work for the sawtooth PD is a quantized version of the continuous random variable described in (16).

4 Numerical results

In order to demonstrate the utility of the results, we consider a standard second order DPLL. The values of N_1, N_2 and N_3 were fixed at 512, 4096 and 128 respectively. The

loop bandwidth is $B_N = 0.2$ and the damping factor is $\frac{1}{\sqrt{2}}$ [1]. The SNR ($\frac{E_s}{N_0}$) was fixed at 5dB for all the trials. The initial pmf was taken as $P(\hat{\theta}_0 = 0, \hat{\Omega}_0 = 0) = 1$.

Here we examine the acquisition performance of the DPLL in detail. First, a constant phase offset of θ_0 =45 degrees, with no frequency offset ($\Omega = 0$) was assumed. Fig. 4 shows the mean of the estimated phase and frequency offsets, i.e., $\mathrm{E}\hat{\theta}_k$ and $\mathrm{E}\hat{\Omega}_k$, with time, along with the linear analysis. Fig. 5 shows the variation of root mean square error, i.e., $\mathrm{E}\{\phi_k\}^2$, with time. It can be seen that the linear analysis underestimates the steady state rmse compared to the Markov analysis for this situation. This is because the linear analysis is not very accurate at the low values of loop SNRs [2]. Fig. 6 shows the pmf of the estimated phase error, ϕ_k during acquisition at various time instants. The Markov analysis was repeated with a phase offset $\theta_0 = 45$ degrees and a normalized frequency offset $\Omega = 0.02\frac{180}{\pi}$ degrees. Fig. 7 again shows the mean of the error in the estimated quantities, $\mathrm{E}\{\theta_k - \hat{\theta}_k\}$ and $\mathrm{E}\{\Omega - \hat{\Omega}_k\}$. Fig. 8 shows the rmse of ϕ_k as a function of time.

It has been noted that coarse quantization produces a biased estimate of the phase offset (θ_k). It has also been noted that the values of N_2 needed for proper phase estimation are typically more than that of N_1, for a fixed B_N. This can be explained as follows. A fixed value of B_N and damping factor give a unique choice of μ_1 and μ_2. Typically the value of μ_2 is much smaller than that of μ_1. For example, the values of μ_1 and μ_2 in these simulations were 0.245 and 0.027 respectively. Thus the gain term associated with μ_2 needs to have higher number of quantization levels. An inadequate value of N_2 can result in a biased estimate of the total phase offset, since the value of $\mu_2\varphi_k$ can go to zero before the frequency offset has been correctly estimated. It has been determined that $N_2 \approx 8N_1$ is usually sufficient for satisfactory performance of the DPLL. It has also been noted that the phase detector typically needs lesser number of quantization levels levels compared to N_1 and N_2.

5 Conclusions

A computationally efficient algorithm for analyzing the quantization effects, in a second order DPLL, using Markov methods, has been presented. It has been determined that the number of quantization levels N_2 needs to be significantly higher than N_1 for satisfactory performance. This algorithm is useful as a design tool for determining the number of quantization levels necessary for satisfactory performance of the DPLL.

References

[1] W. C. Lindsey and C. M. Chie, "A Survey of Digital Phase Locked Loops," *Proc. of the IEEE*, Vol. 69, No. 4, April 1981, pp. 410-431.

[2] W. C. Lindsey, *Synchronization Systems in Communication and Control*, Prentice-Hall, Englewood Cliffs, NJ, 1972.

[3] H. Meyr and G. Ascheid, *Synchronization in Digital Communications*, John Wiley and Sons, NY, 1990.

[4] A. Weinberg and B. Liu, "Discrete Time Analysis of Nonuniform Sampling First and Second-Order Digital Phase Locked Loops," *IEEE Trans. on Comm.* , Vol. COM-22, Feb 1974, pp. 123-137.

[5] E. Cinlair, *Introduction to Stochastic Processes*, Prentice-Hall, Englewood Cliffs, NJ, 1975.

[6] H. J. Larson and B. O. Shubert, *Probabilistic Models in Engineering Sciences*, Vol. II, John Wiley and Sons, NY, 1979.

[7] M. P. Fitz and R. J. Cramer, "A Performance Analysis of a Digital PLL Based MPSK Demodulator," *IEEE Trans. on Comm.* , Vol. COM-43, Feb-March-Apr. 1995, pp. 1192-1201.

[8] W. B. Davenport and W. L. Root, *An Introduction to the Theory of Random Signals and Noise*, IEEE Press, 1987.

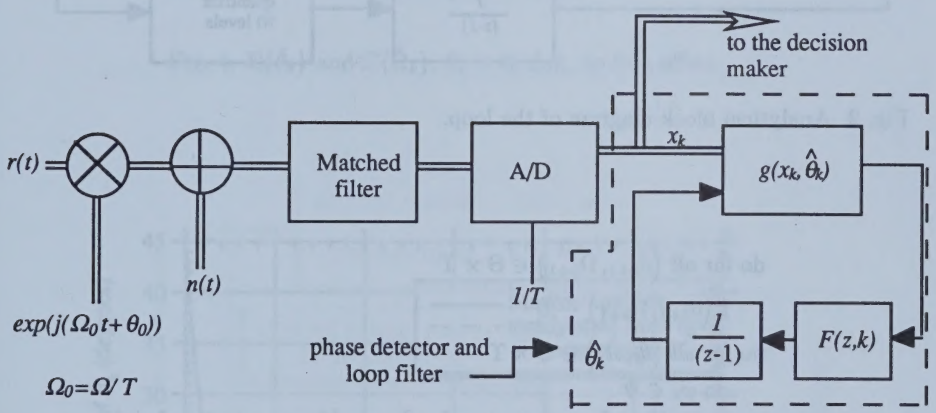


Fig. 1. Block diagram of the DPLL.

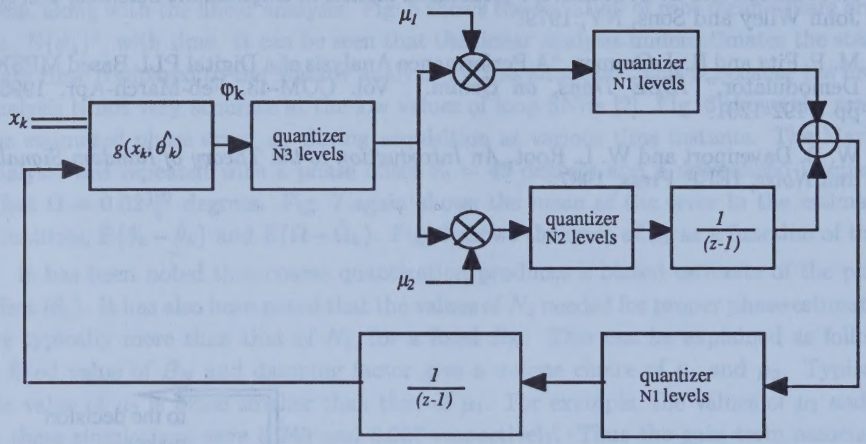


Fig. 2. Analytical block diagram of the loop.

do for all $\left(\hat{\theta}_{k+1}, \hat{\Omega}_{k+1}\right) \in \Theta \times \Upsilon$

$p\left(\hat{\theta}_{k+1}, \hat{\Omega}_{k+1}\right) \leftarrow 0$

do for all $\left(\hat{\theta}_i, \hat{\Omega}_j\right) \in \Theta \times \Upsilon$

do $\varphi_k \in \Psi$

if $\left(\hat{\theta}_i + \hat{\Omega}_j + \mu_1(\varphi_k) = \hat{\theta}_{k+1}\right)$ and $\left(\hat{\Omega}_j + \mu_2(\varphi_k) = \hat{\Omega}_{k+1}\right)$

$p\left(\hat{\theta}_{k+1}, \hat{\Omega}_{k+1}\right) \leftarrow p\left(\hat{\theta}_{k+1}, \hat{\Omega}_{k+1}\right) + p\left(\hat{\theta}_k, \hat{\Omega}_k\right) * p\left(\varphi_k \mid \hat{\theta}_i\right)$

continue

continue

Fig. 3. Pseudo-code for the Algorithm.

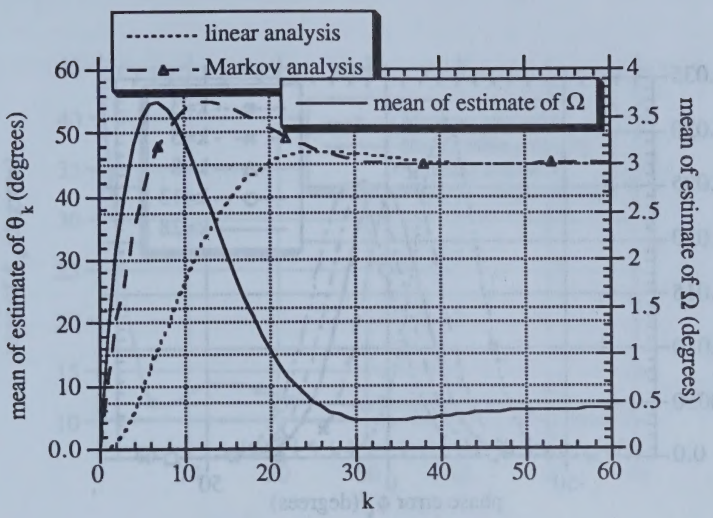


Fig. 4. $\mathrm{E}\{\hat{\theta}_k\}$ and $\mathrm{E}\{\hat{\Omega}_k\}$, $\theta_0 = 45$ deg, no freq offset.

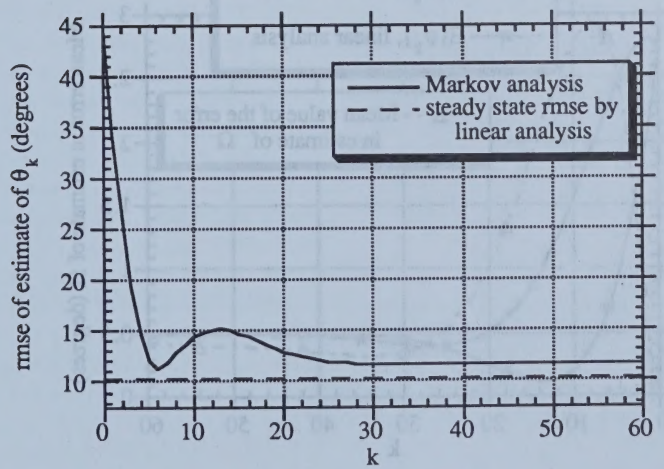


Fig. 5. rmse $\mathrm{E}\{\phi_k\}^2$, $\theta_0 = 45$ deg, no freq offset.

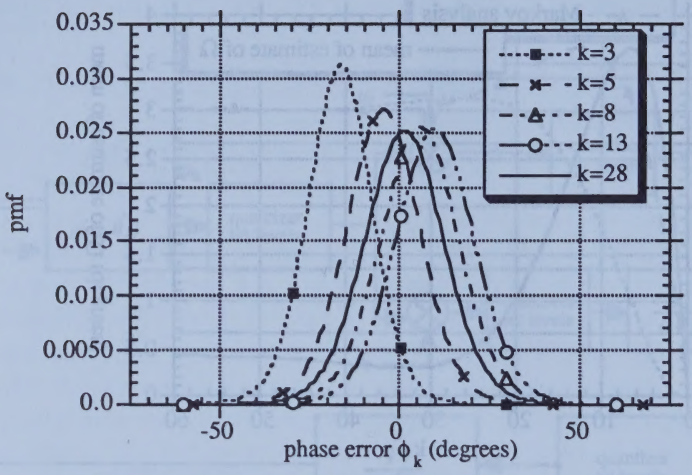


Fig. 6. pmf of ϕ_k during acquisition, $\theta_0 = 45$ deg, no freq offset.

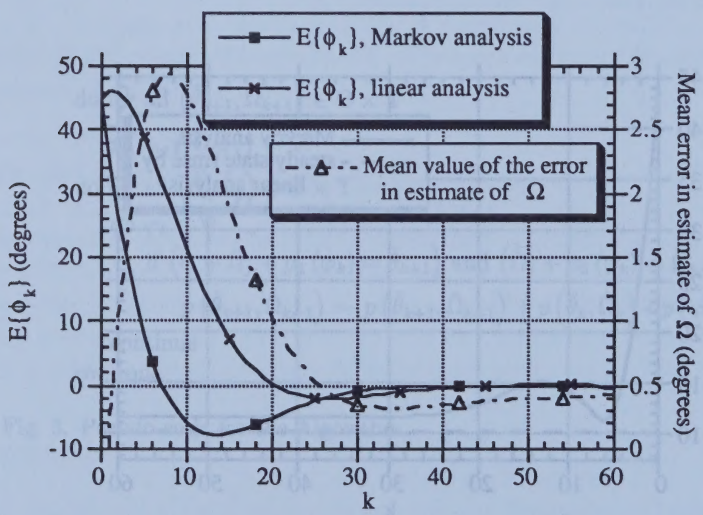


Fig. 7. $E\{\phi_k\}$ and $E\{\Omega - \hat{\Omega}_k\}$, $\theta_0 = 45$ deg, $\Omega = 0.02\frac{180}{\pi}$ deg.

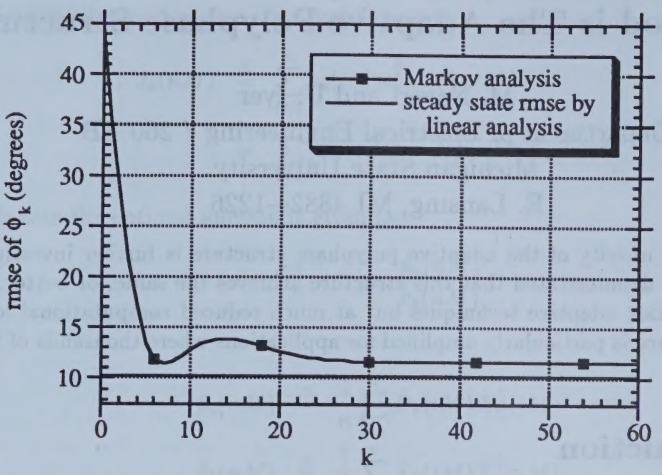


Fig. 8. rmse $E\{\phi_k\}^2$, $\theta_0 = 45$ deg, $\Omega = 0.02\frac{180}{\pi}$ deg.

How Good is The Adaptive Polyphase Structure?

M. Nayeri and U. Iyer
Department of Electrical Engineering / 260 EB
Michigan State University
E. Lansing, MI 48824-1226

Abstract. The novelty of the adaptive polyphase structure is further investigated in this paper. It is demonstrated that this structure achieves the same, or better, performance as other fast adaptive techniques but at much reduced computational load. Its better performance is particularly amplified for applications where thousands of taps are required.

1 Introduction

Sub-band techniques are an effective solution to identify very long impulse responses, specially for acoustic echo cancellation. Their strength lies in their ability to break a given problem into weakly linked sub-problems that can be solved almost independently of one another at faster rates. The forerunner of the sub-band algorithms is the Transform-domain (TD) algorithms which have been studied in [1, 2, 3, 4, 5, 6] to improve the convergence of the adaptive process when the input autocorrelation matrix has very large eigenvalue spread.

It is well-known that Karhunen-Loève Transform (KLT) is the optimal transformation to equalize the eigenvalue spread. However, KLT is a signal-dependent transformation and requires *a priori* knowledge of the second-order statistics of the input. In most practical applications such *a priori* information is not available, or the signal might have time-varying statistics that make a fixed transformation impossible. Under such circumstances, transforms such as the Discrete Fourier Transform (FDLMS), and specially the Discrete Cosine Transform (DCT) have been observed to yield much improved convergence than the regular time domain algorithms [4, 6] since the DCT and, to a less extent, the DFT approach the KLT asymptotically [7]. To further improve the $\mathcal{O}(N \log_2(N))$ computational complexity (per input sample) of these transforms, one may consider a very efficient FFT based block processing scheme called the Fast LMS (FLMS) which was originally introduced by [8] and later analyzed by [9]. In this scheme a FFT's are performed on non-overlapping blocks of data, thereby yielding great savings while compared to the frequency domain LMS (FDLMS) algorithm. However to overcome the aliasing effects of circular convolution, overlap-and-save and a overlap-and-add methods are used. This effectively increases the length of the FFT to be taken to be $2N$ compared to the block size of N. A more efficient block processing algorithm called Adaptive polyphase structure (PPS) was introduced in [10] which will be investigated further in this paper.

Figure 1 shows the adaptive polyphase structure which was analyzed in [10, 11, 12]. In this structure, $x(n)$ denotes the input, $y(n)$ the reconstructed signal, and $d(n)$ the desired signal. In addition, $X_k(nM)$ and $D_k(nM)$ are the kth sub-band signals with accumulated counterparts $\bar{X}_k(nM)$ and $\bar{D}_k(nM)$, respectively. It has been shown that the accumulated

signals of any two distinct bins are asymptotically uncorrelated. This desirable property allows independent adaptation of the adaptive filter coefficients, $\hat{H}_k(nM)$, in each subband of the polyphase structure. The optimal $\hat{H}_k(nM)$ may, for example, be determined by minimizing the least-square-error cost function

$$J_k(nM) \triangleq \sum_{i=1}^{n} e_k(iM)e_k^{\dagger}(iM) \tag{1}$$

$$= \sum_{i=1}^{n} \left| \bar{D}_k(nM) - \hat{H}_k^{\dagger}(nM)\bar{X}_k(nM) \right|^2 \tag{2}$$

in which case the optimal solution is given by

$$\hat{H}_k^{\dagger}(nM) = \frac{\hat{p}_k(nM)}{\hat{r}_k(nM)}, \tag{3}$$

where

$$\hat{p}_k(nM) \triangleq \frac{1}{n}\sum_{i=1}^{n} \bar{D}_k(iM)\bar{X}_k^{\dagger}(iM)$$

$$\hat{r}_k(nM) \triangleq \frac{1}{n}\sum_{i=1}^{n} \bar{X}_k(iM)\bar{X}_k^{\dagger}(iM). \tag{4}$$

The following theorem shows that the algorithm does indeed converge to the samples of the frequency response of the unknown system as intended.

Theorem 1 (Proof: [11]) *Assume that the p.s.d. of input $x(n)$, $S_x(\omega)$, and the frequency response $H_D(\omega)$ of the unknown system are sufficiently smooth functions which belong to $\mathcal{L}([-\pi, \pi])$, then*

$$\lim_{n\to\infty} \frac{\hat{p}_k(nM)}{\hat{r}_k(nM)} = H_D^{\dagger}(\frac{2\pi k}{M}), \quad k \in \{0, 1, \ldots, M-1\} \tag{5}$$

if $S_x(-\frac{2\pi k}{M}) \neq 0$, $k \in \{0, 1, \ldots, M-1\}$.

A straightforward recursive version of the PPS is summarized in Table 1 where the forgetting factor $\lambda \in [0, 1]$ is incorporated in the updates for tracking time-varying systems.

2 Performance Evaluation

In this section, we compare the PPS with some popular algorithms such as LMS, FDLMS, DCTLMS, FLMS, and Fast Transversal Filter (FTF). The strength of LMS is its simplicity but its weakness is its slow convergence while using severely colored inputs. The DCTLMS and FDLMS are faster than LMS when the input is severely colored but are computationally expensive. FTF a variant of the Recursive Least Square algorithms is computationally more efficient that RLS, but is complex and unstable. The FLMS algorithm offers both computational savings and better performance than LMS while using colored inputs. We will compare the performance of PPS with these algorithms, especially for echo cancellation applications. In order to evaluate the performance of PPS while suppressing echoes, a measure of echo suppression, the Echo Return Loss Enhancement (ERLE), defined as

- Form the input blocks at instant nM as

$$\begin{aligned} \boldsymbol{x_B}(n) &= [x(nM), x(nM-1), \cdots, x((n-1)M+1)] \\ \boldsymbol{d_B}(n) &= [d(nM), d(nM-1), \cdots, d((n-1)M+1)] \end{aligned}$$

  where $x(nM)$ is the input and $d(nM)$ is the desired output

- Find the FFT of the input blocks

$$[X_0(nM), X_1(nM), \cdots, X_{M-1}(nM)] = FFT(\boldsymbol{x_B}(n))$$

$$[D_0(nM), D_1(nM), \cdots, D_{M-1}(nM)] = FFT(\boldsymbol{d_B}(n))$$

- for $i = 0, 1, \ldots, M-1$ do

  1. Form the accumulated outputs

$$\begin{aligned} \bar{X}_i(nM) &= \lambda \bar{X}_i((n-1)M) + X_i(nM) \\ \bar{D}_i(nM) &= \lambda \bar{D}_i((n-1)M) + D_i(nM) \end{aligned}$$

  2. Update the inverse auto-correlation and cross-correlations

$$\begin{aligned} \pi(nM) &= \bar{X}_i^{\dagger}(nM) P((n-1)M) \\ \kappa(nM) &= P((n-1)M)\bar{X}_i(nM)/(\lambda + \pi(n)\bar{X}_i(nM)) \\ e_i(nM) &= \bar{D}_i(nM) - \hat{H}_i^{\dagger}(nM)\bar{X}_i(nM) \\ P(nM) &= \frac{1}{\lambda}\{P((n-1)M) - \kappa(nM)\pi(nM)\} \end{aligned}$$

  3. Then

$$\hat{H}_i^{\dagger}(nM) = \hat{H}_i^{\dagger}((n-1)M) + \kappa(nM)e_i^{\dagger}(nM)$$

- Obtain the impulse response estimate $\hat{h}(n)$ as

$$\hat{h}(n) = IFFT\{[\hat{H}_0(nM), \hat{H}_1(nM), \cdots, \hat{H}_{M-1}(nM)]\}$$

Table 1: Recursive PPS algorithm.

$$ERLE = 10log\frac{E\{y^2(n)\}}{E\{e^2(n)\}}, \quad (6)$$

is used in some experiments, where $y(n)$ is the desired output, and $e(n)$ is the output error. An ERLE greater that 20dB is expected for echo cancellation [13].

2.1 Convergence

In the first scenario the unknown system is a 512 tap FIR filter. Two simulations were performed, one with a low pass input and the other with a high pass input, where the stop-band attenuation in both cases are around 40dB. Figures 2 and 3 portray the results of this comparison. It is clear that the performance of LMS is the worst and that DCTLMS gives the best performance. PPS does a reasonable job, and gives an ERLE of more than 20dB. At the same time PPS is computationally more efficient, requiring a 512 point FFT every 512 samples, while DCTLMS and FDLMS require the corresponding 512 point transform for every sample.

In the second scenario the impulse response of a real conference room room of dimensions 7mx8mx2.5m (Figure 4) is the unknown system, and the input is a speech signal (a male voice reading a passage from a book). Figure 5 compares the echo suppression achieved by the different algorithms. The superiority of PPS over LMS is self-evident. Its performance is comparable with that of FDLMS and DCTLMS, but note that FDLMS and DCTLMS are much more computationally intensive. After 50,000 samples while FDLMS and DCTLMS perform 50,000 of their corresponding 4096-point transform, PPS performs only 12 FFTs.

PPS, DCTLMS and FDLMS fare much better than LMS, when the input is ill-conditioned. DCTLMS gives a very good performance with speech and low pass signals, while FDLMS fares well for high pass inputs. Notice that when the order of the system increases PPS performs at least as well as the other algorithms. This along with the fact that it is $\mathcal{O}(M-1)$ times more efficient than both DCTLMS and FDLMS, makes it the best choice for echo cancellation. PPS offers a steady performance, achieving the ERLE of 20dB after only 10 blocks of input have been processed, for almost all inputs and all orders of the system.

As a further test of the performance of PPS we compare PPS to FLMS. Figure 6 shows how PPS outperforms FLMS. The order of the system used in this case was 256 and the input was a low pass signal with a cut-off frequency of 0.25π. Not only does PPS perform better, it is computationally very efficient compared to FLMS.

Note that the memory allocation for the covariance matrix (4096×4096 in this case) make RLS sluggish and impractical as verified by our simulations. But its fast stable version, SFTF, performs very well compared to PPS in the non-tracking mode. In fact, one may conclude that SFTF outperforms PPS. As will be shown in the next section, stability of SFTF is not guaranteed even in the presence of slowly varying systems.

2.2 Computational Complexity

The strength of PPS has been its computational efficiency. Not only does it allow parallel processing but also improves efficiency by being a block processing algorithm. This feature makes it ideal for applications, involving the identification of very long impulse responses, such as severe echo cancellation in telephones and teleconferences.

Algorithm	$\otimes$ per sample	N = 128	N = 512	N = 1024
LMS	2N	256	1024	2048
FTF	8N	1024	4096	8192
PPS	$6\log_2(N/2)+8$	44	56	62
FLMS	$10\log_2(2N)+16$	96	116	126

Table 2: Comparison of computational complexities of LMS, FTF and PPS.

The problem of echo cancellation involves the identification of a system whose impulse response is as long as the round trip time of the call [14, 15]. With increasing distances over which a call is made, the effectiveness of a canceller is limited by its computational efficiency. In addition, the non-stationary nature of the signals and their sample correlation inhibit the effectiveness of many algorithms. Table 2 compares the computational complexities of the LMS, FTF, PPS, and FLMS. Notice that PPS enjoys a decisive computational advantage over the other three algorithms.

We next evaluate the ability of PPS to track time-varying systems.

2.3 Parameter Tracking

With the introduction of the forgetting factor PPS has the ability to track time varying systems. To verify its ability to track time varying systems, we test its tracking properties first with slowly varying systems and then with fast variations.

In the first case, sinusoidal variations are introduced into one parameter of a 512 tap FIR filter. The input in this case is a high pass signal with a cut-off frequency of 0.5π. Figure 7 shows how PPS outperforms LMS in tracking the time-varying parameter. In order to model systems with larger variations, ramp variations are introduced into 8 parameters of a 128 tap FIR filter. In Figure 8 only three of the parameters have been plotted for convenience. Notice that PPS shows a much faster and better tracking of the unknown parameters.

We next compare the performance of PPS with FTF. FTF exhibits all the fast convergence properties of a Least Squares based algorithm, but is very complex and inherently unstable. In order to make it more stable numerous rescue and restart operations have been prescribed which add to the complexity of the algorithm. Nevertheless it is a yard stick for comparing both the speed of convergence and the computational efficiency. We shall address the issue of computations in a later section. Here we present the case where the instability of this algorithm in case of time varying systems is presented.

In the first comparison shown in Figure 9, PPS performs marginally better that FTF. In this case the length of the unknown system was 16 and saw-tooth type variations were introduced into on the parameters. The input signal was low pass with a cut-off frequency of 0.1π. When the order of the unknown system is increased to 128 it can be seen from Figure 10 that FTF becomes unstable around the sample 2500. In spite of the presence of rescue and restart operations FTF, goes unstable. Further modifications have been proposed in [16] to overcome this problem

It has been noticed that PPS exhibits excellent tracking capabilities. It exhibits a very fast transient state, and once it settles in a neighborhood of the true parameter, it achieves tracking even under adverse input conditions. The PPS elevates its performance beyond the reach of the LMS, for both slow and fast-varying systems.

3 Conclusion

We have considered an asymptotically alias-free parallel structure in which the DFT filter bank polyphase decomposition allows multi-rate processing. The zooming effect, as a result of using accumulators in each band, is the distinguishing feature of this scheme. This implies that the bands would become asymptotically uncorrelated, which accommodates independent adaptations in each sub-band. That is, $\hat{H}_k(nM)$, given by (3), converges to $H_D(\frac{2\pi k}{M})$ as $n \rightarrow \infty$ provided sufficient number of bands are used and the input is persistently exciting.

Through several experiments it was shown that PPS outperforms the LMS and FDLMS algorithms in most applications involving severe input coloring. The tracking capability is yet another strong attribute of PPS which was experimentally shown against the LMS and FTF. The great computational saving afforded by PPS as compared to LMS, FTF and FLMS is evident from Table 2.This coupled with its good performance under different SNRs [12] and its effectiveness in identifying long impulse responses, especially under adverse input coloring and changing environment, makes this structure a viable alternative to available methods, including IIR techniques.

References

[1] N. Bershad and P. Feintuch, "Analysis of frequency domain adaptive filter," *Proc. IEEE*, vol. 67, pp. 1658–1659, Dec. 1979.

[2] R. Bitmead and B. Anderson, "Adaptive frequency sampling filters," *IEEE Trans. on Circuits and Systems*, vol. 28, pp. 610–615, Jun. 1981.

[3] M. Dentino, J. McCool, and B. Widrow, "Adaptive filtering in the frequency domain," *Proc. IEEE*, vol. 66, pp. 1658–1659, Dec. 1978.

[4] J. Murphy, *A Comparison of Transform Domain Adaptive Filters, with Emphasis on the Hartley Transform.* PhD thesis, University of Illinois, Urbana, 1987.

[5] S. Narayan and A. Peterson, "Frequency domain LMS algorithm," *Proc. IEEE*, vol. 69, pp. 124–126, Jan. 1981.

[6] S. Narayan, A. Peterson, and M. Narasimha, "Transform domain LMS algorithm," *IEEE Trans. on Acoustics, Speech and Signal Processing*, pp. 609–615, Jun 1983.

[7] S. Haykin, *Adaptive Filter Theory.* Prentice Hall, 1996.

[8] J. E.R. Ferrara, "Fast implementation of LMS adaptive filters," *IEEE Trans. on Acoust., Speech, and Signal Processing*, vol. ASSP-28, pp. 474–475, Aug 1980.

[9] J. Lee and C. Un, "Performance analysis of frequency-domain adaptive filters using overlap-add methods," *IEEE Trans. on Circuits and Systems*, vol. 36, pp. 173–189, Feb 1989.

[10] U. Iyer, H. Ochi, and M. Nayeri, "IIR subband adaptive filtering," *Midwest Symp. on Sytems and Circuits*, vol. 1, pp. 125–128, Aug 1993.

[11] U. Iyer, M. Nayeri, and H. Ochi, "A polyphase based adaptive structure for adaptive filtering and tracking," *IEEE Trans. on Circuits and Systems II*, vol. 43, pp. 220–233, Mar. 1996.

[12] U. Iyer, *A Polyphase-Based Alias-Free Structure For Adaptive Filtering and Tracking.* PhD thesis, Michigan State University, East Lansing, Michigan, 1996.

[13] H. Fan, *New Adaptive IIR Filtering Algorithms.* PhD thesis, University of Illinois, Urbana, 1985.

[14] G. Maral and M. Bousquet, *Satellite Communications Systems.* John Wiley and Sons, 1986.

[15] B. Widrow and S. Stearns, *Adaptive Signal Processing.* Prentice Hall, 1985.

[16] D. Slock, C. Luigi, H. Lev-Ari, and T. Kailath, "Modular and numerically stable fast transversal filters for multichannel and multiexperiment RLS," *IEEE Trans. on Signal Processing*, vol. 40, pp. 784–793, Apr. 1992.

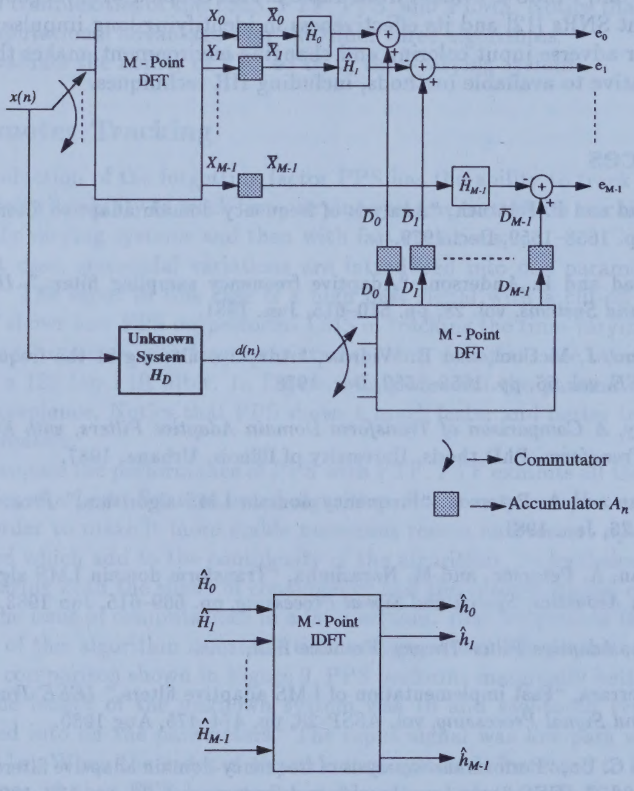


Figure 1: Adaptive polyphase structure.

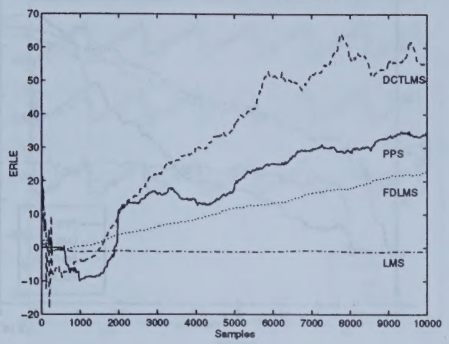


Figure 2: Comparison of learning curves for LMS, FDLMS, DCTLMS and PPS when the input is a low pass colored noise. The unknown system is a 512 tap FIR filter.

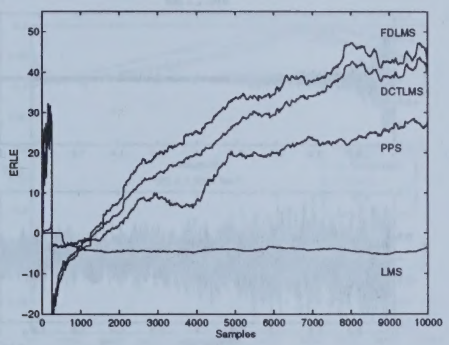


Figure 3: Comparison of learning curves for LMS, FDLMS, DCTLMS and PPS when the input is a high pass colored noise. The unknown system is a 512 tap FIR filter.

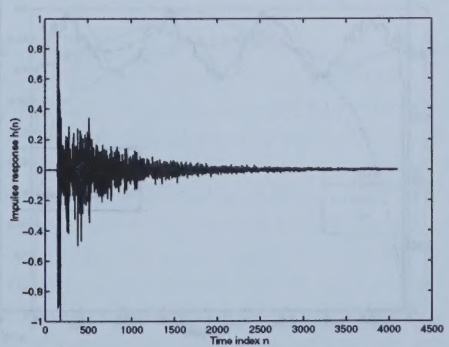


Figure 4: Impulse response of a room.

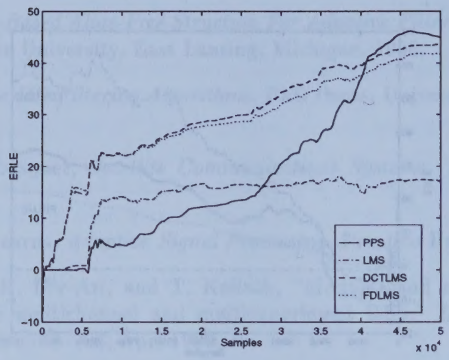


Figure 5: Comparison of learning curves for LMS, FDLMS, DCTLMS, and PPS when the input is a speech signal. The unknown system is the 4096 tap impulse response of a conference room.

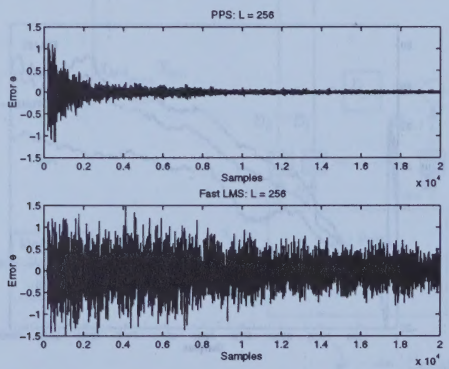


Figure 6: Comparison of Performance of PPS and FLMS. N = 256 and input is a low pass signal with cut-off of 0.25π.

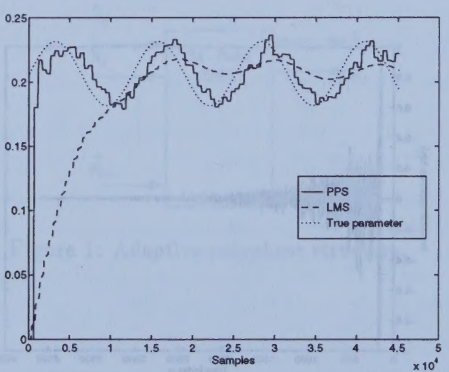


Figure 7: Tracking behavior of LMS and PPS with slow variations in one parameter of a 512-tap FIR system ($\lambda = 0.45$).

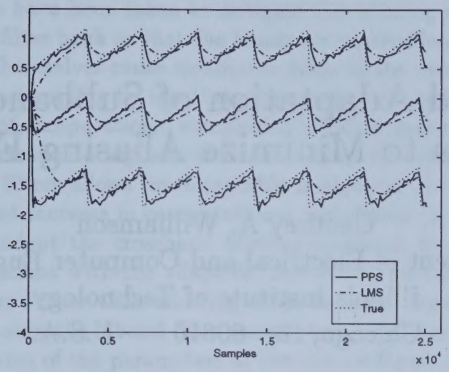


Figure 8: Tracking behavior of LMS and PPS with variations in 8 parameters of a 128-tap FIR system ($\lambda = 0.35$).

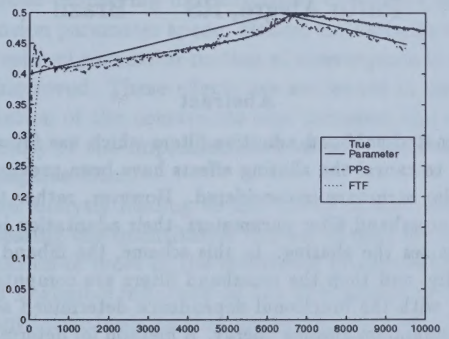


Figure 9: Tracking behavior of FTF and PPS with variations in 1 parameter of a 16-tap FIR system ($\lambda_{pps} = 0.95, \lambda_{ftf} = .985$).

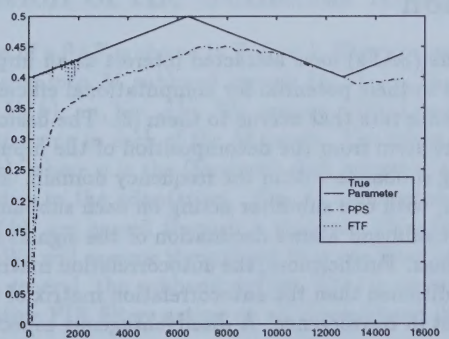


Figure 10: Tracking behavior of FTF and PPS with variations in 1 parameter of a 128-tap FIR system ($\lambda_{pps} = 0.9, \lambda_{ftf} = .9$). Notice that FTF goes unstable around sample 2500.

Constrained Adaptation of Subband Adaptive Filters to Minimize Aliasing Effects

Geoffrey A. Williamson
Department of Electrical and Computer Engineering
Illinois Institute of Technology
Chicago, IL 60616 U.S.A.

Sergio Zimmermann
Department of Electrical Engineering
Universidade Federal do Rio Grande do Sul
Porto Alegre, RS Brazil

Abstract

Critically decimated subband adaptive filters which use filtering between subbands in an effort to cancel the aliasing effects have been previously proposed. In this paper, a similar structure is considered. However, rather than allowing free adaptation of the crossband filter parameters, their adaptation is constrained in a fashion that minimizes the aliasing. In this scheme, the inband filter parameters are adapted directly, and then the crossband filters are computed as functions of the inband filters, with the functional dependence determined so that the overall adaptive filter has minimum aliasing energy. A method for determining which crossband filters are significant is implied by the constraint equations, and guidelines for which crossband filters should be retained are given.

1 Introduction

Subband adaptive filters (SAFs) have attracted interest as an implementation structure for adaptive filters due to their potential for computational efficiency as well as the improvements in convergence rate that accrue to them [2]. The basic mechanisms for these performance advantages stem from the decomposition of the input signals into approximately non-overlapping subbands within the frequency domain. The adaptive filter acts on the subband signals, with one sub-filter acting on each subband signal. The reduced bandwidth within each subband allows decimation of the signals lying therein, yielding the computational savings. Furthermore, the autocorrelation matrices of the subband regressors are better conditioned than the autocorrelation matrix of the fullband regressor; hence, the improvement in convergence. A disadvantageous aspect of subband adaptive filters, however, is that the overlap between adjacent subbands introduces aliasing error into the adaptive filter output, thereby degrading performance, especially if the subbands are critically decimated [1].

Several approaches have been taken to mitigate this aliasing error. One approach is to adjust the analysis filter bank so that the bands are nonoverlapping, but the resulting "spectral holes" can themselves cause significant error in the overall response [3]. Subcritical sampling (downsampling by less than the number of bands) reduces the aliasing error, but at the cost of computational complexity [4]. One may also reduce the aliasing error by allowing for cross-filtering between subbands in the adaptive filter. The addition of the crossband filters allows for acceptable mitigation of the aliasing error while incurring only a modest increase in computational complexity [1].

In this paper, we adopt the crossband filtering approach to elimination of aliasing error. However, in contrast with the adaptive filter of [1], we constrain the adaptation of the crossband filters rather than allowing them to adapt freely. To implement the constraints, we adapt only the inband filter parameters, and then specify the crossband filter parameters in terms of the parameters of the inband filters. The crossband parameters are set so that the resulting overall filter has the minimum aliasing error among all filters with the given inband filters values. We conduct the adaptation of the inband filters while taking into account the associated effects of the crossband filters.

This approach confers the following advantage to constrained subband adaptive filters (CSAFs) relative to those employing unconstrained crossband filters. The adaptation occurs in a lower dimension parameter space. Hence, the variance error (or "excess mean-square error") due to residual parameter motion at convergence is reduced. Furthermore, convergence speed is improved. These effects are evidenced in our simulation trials. Of course, the implementation of the constraints also increases the computational burden, but the overall complexity of the implementation of the CSAF remains comparable to that of the fullband adaptive filter.

A by-product of the analysis leading to the CSAF is that the constraint equations provide a means of assessing the quantitative impact of each crossband filter. In [1], the authors suggest elimination of most of the crossband filters, as their experience showed that most of these had negligible influence on the elimination of the aliasing error. Our analysis provides a quantitative basis for this elimination of crossband filters, and provides a design methodology for deciding which crossband filters to keep, given a specification of the analysis and synthesis filter banks used in the implementation.

2 Configuration of the Subband Adaptive Filter

The basic configuration of a SAF is shown in Figure 1. Two analysis filter banks split the adaptive filter input $u(n)$ into M subband signals $U_0(k), \ldots, U_{M-1}(k)$ and the desired filter output $y(n)$ into $Y_0(k), \ldots, Y_{M-1}(k)$. We assume critical decimation, with subsampling by a factor M occurring in each of the M bands. The adaptive filter is represented in the figure by $C(q,k)$, where q is the (forward) shift operator and where k represents the time dependence due to the adaptation. The M subband outputs of the adaptive filter, $\widehat{Y}_0(k), \ldots, \widehat{Y}_{M-1}(k)$, are passed through a synthesis filter bank to reconstruct the fullband output $\hat{y}(n)$. We will assume throughout that the filter banks are perfect reconstruction (PR) [5]. In general, the subband output $\widehat{Y}_i(k)$ is constructed as combination of the outputs of adaptive FIR filters acting on all the subband inputs:

$$\widehat{Y}_i(k) = \sum_{j=0}^{M-1} C_{i,j}(q,k) U_j(k).$$

Here, $C_{i,i}(q,k)$ is the inband filter acting in band i; $C_{i,j}(q,k)$ with $i \neq j$ is the crossband filter taking band j to band i.

In order to obtain comparable performance to a fullband FIR filter of order N, each inband and crossband filter would be implemented as an FIR filter with a length $L_f = N/M$. Note that this indicates that $C(q,k)$ contains within it $L_f M^2 = MN$ adjustable parameters. Due to the down and up sampling in the filter, the overall impulse response of the SAF is periodically time-varying with period M when its parameters are held constant. Therefore, the MN degrees of freedom in correspond in a rough sense to a periodically time-varying impulse response of length N with period M. Note that the correspondence is rough, because the filter banks $\{H_i(z)\}$ and $\{F_i(z)\}$ will introduce some lag in the filter response, so that the (periodically time-varying) impulse response of the filter from $u(n)$ to $\hat{y}(n)$ will actually have length somewhat greater than N.

Although the SAF will have a periodically time-varying response, in most situations one desires the adaptive filter to model some real world process or optimal filter which in fact has a time-invariant response. This being the case, the desired behavior of the SAF is to adapt to a setting for which this time invariance is achieved. The space of all linear time-invariant systems with impulse response of length N is an N-dimensional space, yet our SAF has (in general) MN parameters. It is hence overparametrized by $(M-1)N$ parameters!

In most implementations, not all crossband filters will be implemented. In fact, the first SAF designs possessed only inband adaptive filters, for a total of N parameters, and did not include any of the crossband filters, which account for the remaining $(M-1)N$ parameters of the general SAF. As we noted, the drawback of eliminating the crossband filters is that aliasing effects then severely compromise the overall SAF performance. Still, it has been observed that the full flexiibility of including all crossband filters is unnecessary. The SAF of [1] uses only adjacent crossband filters ($C_{i,j} = 0$ for $i \notin \{j-1, j, j+1\}$), and furthermore restricts the length of the crossband filters that are retained to approximately $L_f/3$. The reason that this is possible is that if the analysis filter bank does a reasonable separation of the bands, then correlation between non-adjacent bands is negligible. If we take the view that the inband filters carry the burden of modeling the essence of the relationship between u and y, then the crossband filters' task is to eliminate the effects of correlation between the bands, which shows up as aliasing error. With neglible correlation between the bands, only adjacent crossband filters are needed to do this, and even then only a limited number of parameters are required.

In this paper, we will take the view that only N degrees of freedom are needed to model the length N time-invariant filter response. We adapt within an N-dimensional subspace of the full MN-dimensional parameter space by constraining the overall response to be time-invariant. We parametrize this space via the N inband filter parameters, and use these to define crossband filter parameters. This approach is dubbed a Constrained SAF, as the adaptation is constrained to lie within that N-dimensional space that is most closely aligned with the space of linear time-invariant systems.

3 Constraining the Crossband Filters

Consider that the parameters of $C(q,k)$ in Fig. 1 are constant, and let its transfer function be denoted by $C(z) = [C_{i,j}(z)]$. The transfer function between $u(n)$ and $\hat{y}(n)$ may then

be written as

$$\widehat{Y}(z) = \frac{1}{M} \begin{bmatrix} \sum_{i,j=0}^{M-1} F_i(z) C_{i,j}(z^M) H_j(z) \\ \sum_{i,j=0}^{M-1} F_i(z) C_{i,j}(z^M) H_j(Wz) \\ \vdots \\ \sum_{i,j=0}^{M-1} F_i(z) C_{i,j}(z^M) H_j(W^{M-1}z) \end{bmatrix}^T \mathbf{U}(z), \tag{1}$$

where

$$\mathbf{U}(z) = \begin{bmatrix} U(z) & U(Wz) & \cdots & U(W^{M-1}z) \end{bmatrix}^T$$

and where $W = e^{j2\pi/M}$. To constrain the response to exclude any aliasing components, we must have

$$\sum_{i,j=0}^{M-1} F_i(z) C_{i,j}(z^M) H_j(W^\ell z) \tag{2}$$

equal to zero for $\ell = 1, \ldots, M-1$. (The expression in (2) for $\ell = 0$ is the unaliased component.) This is an ideal goal that unfortunately cannot be achieved. However, we will demonstrate one way to make the set of alias components in (2) collectively small. To do this , we will first derive the relationship of the filter parameters, inband as well as crossband, to the alias components. Then, we develop expressions for the coefficients of the $C_{i,j}(z)$, $i \neq j$, in terms of the coefficients of $C_{i,i}(z)$ for $i = 0, \ldots M$ and the coefficients of the products $F_i(z)H_j(W^\ell z)$ for $i, j = 0, \ldots, M-1$ and $\ell = 1, \ldots, M-1$, in order that these alias components are minimized.

Denote the coefficients of the $F_i(z)H_j(W^\ell z)$ product by $p^\ell_{i,j,k}$ where

$$F_i(z)H_j(W^\ell z) = p^\ell_{i,j,0} + p^\ell_{i,j,1} z^{-1} + \cdots + p^\ell_{i,j,2L_s-1} z^{-2L_s-1}, \tag{3}$$

where L_s is the length of the filters in the filter banks. We use L_s to denote the length of the constituent FIR filters in the filter banks. Now define the $(L_s+1)M \times L_s$ sylvester-like matrix

$$P^\ell_{i,j} = \begin{bmatrix} p^\ell_{i,j,0} & 0 & 0 \\ \vdots & \vdots & \vdots \\ \vdots & 0 & \\ p^\ell_{i,j,2L_s-1} & p^\ell_{i,j,0} & \\ 0 & \vdots & \\ \vdots & p^\ell_{i,j,2L_s-1} & \vdots \\ & 0 & 0 \\ & \vdots & p^\ell_{i,j,0} \\ \vdots & \vdots & \vdots \\ 0 & 0 & p^\ell_{i,j,2L_s-1} \end{bmatrix}, \tag{4}$$

where the stack of zeros atop the second column is of height M and the stack of zeros atop the last column is of height $(L_s - 1)M$. Further set

$$P^\ell_c = \begin{bmatrix} P^\ell_{0,1} & \cdots & P^\ell_{0,M-1} & P^\ell_{1,0} & P^\ell_{1,2} & \cdots & P^\ell_{1,M-1} & \cdots & P^\ell_{M-1,0} & \cdots & P^\ell_{M-1,M-2} \end{bmatrix} \tag{5}$$

and

$$P_i^\ell = \left[\begin{array}{cccc} P_{0,0}^\ell & P_{1,1}^\ell & \cdots & P_{M-1,M-1}^\ell \end{array} \right].$$

Then defining

$$\mathbf{c}_{i,j} = \left[\begin{array}{cccc} c_{i,j,0} & c_{i,j,1} & \cdots & c_{i,j,L_f-1} \end{array} \right]^T \tag{6}$$

as the vector of parameters in the filter between bands j and i, we may form vectors consisting of all the inband filter parameters and all the crossband filter parameters as

$$\theta^{(i)} = \left[\begin{array}{cccc} \mathbf{c}_{0,0}^T & \mathbf{c}_{1,1}^T & \cdots & \mathbf{c}_{M-1,M-1}^T \end{array} \right]^T$$

$$\theta^{(c)} = \left[\begin{array}{ccccccccccc} \mathbf{c}_{0,1}^T & \cdots & \mathbf{c}_{0,M-1}^T & \mathbf{c}_{1,0}^T & \mathbf{c}_{1,2}^T & \cdots & \mathbf{c}_{1,M-1}^T & \cdots & \mathbf{c}_{M-1,0}^T & \cdots & \mathbf{c}_{M-1,M-2}^T \end{array} \right]^T$$

respectively. With this notation in hand, we may then express (2) as

$$\left[\begin{array}{cccc} 1 & z^{-1} & \cdots & z^{-(2L_s+N-1)} \end{array} \right] \left(P_c^\ell \theta^{(c)} + P_i^\ell \theta^{(i)} \right).$$

One approach to making the alias components simultaneously small is to force the concatenated vector of coefficients

$$P_c \theta^{(c)} + P_i \theta^{(i)}$$

to be small in a Euclidean norm sense, where we have set

$$P_c = \left[\begin{array}{c} P_c^1 \\ P_c^2 \\ \vdots \\ P_c^{M-1} \end{array} \right]; \qquad P_i = \left[\begin{array}{c} P_i^1 \\ P_i^2 \\ \vdots \\ P_i^{M-1} \end{array} \right].$$

If we assume that the $\theta^{(i)}$ coefficients are given (as they will be by the adaptive algorithm), then we may find the choice of $\theta^{(c)}$ given $\theta^{(i)}$ to minimize the vector of alias component coefficients as

$$\theta^{(c)} = -(P_c^H P_c)^{-1} P_c^H P_i \theta^{(i)}. \tag{7}$$

Clearly, in expresion (7), we must have the matrix $P_c^H P_c$ be invertible. Note that P_c, as given in (5) using (4) and (3), depends only on the analysis and synthesis filter banks, and so this invertibility requirement does not change with adaptation.

Zimmermann notes in [6] that in all cases he examined with perfect reconstruction filter banks, $P_c^H P_c$ was indeed invertible. We expect that one may be able to formally establish invertibility using the sylvester-like structure of P_c^ℓ and the properties of perfect reconstruction filter banks. However, this to date remains an open question.

We will later discuss adaptive algorithms that adjust only $\theta^{(i)}$, and then define $\theta^{(c)}$ via (7). In this way, the SAF is constrained to adapt within the parameter space for which the alias components are minimized for each choice of $\theta^{(i)}$.

4 Simplifying the Constraints

As we noted earlier, in implementation such as in [1], not all the crossband filters are utilized. In constrast, the constrained SAF noted above developed above does employ all available crossband filters. We now address the question of how to choose which crossband filters can be neglected without compromising the mitigation of the alias component effects.

Let the matrix $-(P_c^H P_c)^{-1} P_c^H P_i$ in (7) be denoted by Γ. The natural block structure for Γ is

$$\Gamma = \begin{bmatrix} \Gamma_{1,0}^0 & \Gamma_{1,0}^1 & \cdots & \Gamma_{1,0}^{M-1} \\ \Gamma_{2,0}^0 & \Gamma_{2,0}^1 & \cdots & \Gamma_{2,0}^{M-1} \\ \vdots & \vdots & & \vdots \\ \Gamma_{M-2,M-1}^0 & \Gamma_{M-2,M-1}^1 & \cdots & \Gamma_{M-2,M-1}^{M-1} \end{bmatrix} \tag{8}$$

where $\Gamma_{i,j}^{\ell}$ denotes the linkage between the inband parameters of band ℓ to the crossband parameters of the filter from band j to band i. Recall that in [1], nonadjacent crossband filters were neglected. Using Γ, we present a quantitative technique for determining which crossband filters can be neglected in an implementation. For crossband filters whose effects are insignificant, the corresponding $\Gamma_{i,j}^{\ell}$ terms should be negligible. The following puts a quantitative measure to what is meant by negligible.

We have

$$\theta = \begin{bmatrix} \theta^{(i)} \\ \theta^{(c)} \end{bmatrix} = \begin{bmatrix} \theta^{(i)} \\ \Gamma\theta^{(i)} \end{bmatrix}.$$

Let $\widehat{\Gamma}$ be equal to Γ, but with rows corresponding to some set of crossband parameters set to zero. For instance, setting the blocks in (8) whose subscript is i,j to zero connotes elimination of the entire crossband filter from band j to band i.[1] Define

$$\hat{\theta} = \begin{bmatrix} \theta^{(i)} \\ \widehat{\Gamma}\theta^{(i)} \end{bmatrix} = \begin{bmatrix} \theta^{(i)} \\ \hat{\theta}^{(c)} \end{bmatrix}.$$

This denotes an approximation of the desired SAF implementation, using a limited set of crossband filters.

With respect to the signals depicted in Fig. 1, we may write

$$\begin{bmatrix} \widehat{Y}_0(k) \\ \vdots \\ \widehat{Y}_{M-1}(k) \end{bmatrix} = \Phi(k) \begin{bmatrix} \theta^{(i)} \\ \theta^{(c)} \end{bmatrix}, \tag{9}$$

where we assume fixed parameters. The matrix $\Phi(k)$ is given by

$$\Phi(k) = \begin{bmatrix} \Phi^{(i)}(k) & \Phi_0^{(c)}(k) & \Phi_1^{(c)}(k) & \cdots & \Phi_{M-1}^{(c)}(k) \end{bmatrix} \tag{10}$$

where $\Phi^{(i)}(k)$ and $\Phi_j^{(c)}(k)$ are defined by

$$\Phi^{(i)}(k) = \begin{bmatrix} \phi_0(k) & & & \\ & \phi_1(k) & & \\ & & \ddots & \\ & & & \phi_{M-1}(k) \end{bmatrix}$$

and

$$\Phi_j^{(c)}(k) = \begin{bmatrix} \phi_j(k) & & & & \\ & \ddots & & & \\ & & \phi_j(k) & 0 & \\ & & 0 & 0 & \\ & & 0 & \phi_j(k) & \\ & & & & \ddots \\ & & & & \phi_j(k) \end{bmatrix} \tag{11}$$

[1] To eliminate some subset of the parameters in crossband filters from band j to band i, one would set to zero the corresponding subset of the $\Gamma_{i,j}^{\ell}$ blocks in Γ for each ℓ.

and in turn

$$\phi_j(k) = \left[\begin{array}{ccc} U_j(k) & \cdots & U_j(k-L_f+1) \end{array} \right]. \tag{12}$$

In (11), $\Phi_j^{(c)}(k)$ is a matrix with M rows and $M-1$ block columns. The j^{th} row of $\Phi_y^{(c)}(k)$ is the row consisting entirely of zeros. If $\{u(n)\}$ is white and the filter banks are of a normalized-lossless perfect reconstruction design [5], then

$$E\{\Phi^T(k)\Phi(k)\} = I$$

so that from (9) we have

$$\sigma_{\widehat{Y}}^2 \stackrel{\text{def}}{=} E\left\{ \left\| \left[\begin{array}{c} \widehat{Y}_0 \\ \vdots \\ \widehat{Y}_{M-1} \end{array} \right] \right\|^2 \right\} = \left\| \left[\begin{array}{c} \theta^{(i)} \\ \theta^{(c)} \end{array} \right] \right\|^2. \tag{13}$$

Defining

$$\left[\begin{array}{c} \widetilde{Y}_0 \\ \vdots \\ \widetilde{Y}_{M-1} \end{array} \right] = \Phi(k) \left[\begin{array}{c} \hat{\theta}^{(i)} \\ \theta^{(c)} \end{array} \right] - \Phi(k) \left[\begin{array}{c} \theta^{(i)} \\ \hat{\theta}^{(c)} \end{array} \right]$$

as the difference between $\widehat{Y}$ formed using the desired parameters and $\widehat{Y}$ form using an approximation of the desired parameters (obtained by elimination of selected crossband filters), we have

$$\sigma_{\widetilde{Y}}^2 \stackrel{\text{def}}{=} E\left\{ \left\| \left[\begin{array}{c} \widetilde{Y}_0 \\ \vdots \\ \widetilde{Y}_{M-1} \end{array} \right] \right\|^2 \right\} = \left\| \left[\begin{array}{c} 0 \\ \theta^{(c)} - \hat{\theta}^{(c)} \end{array} \right] \right\|^2 = \left\| \theta^{(c)} - \hat{\theta}^{(c)} \right\|^2. \tag{14}$$

in analogy with (13). We may then use the ratio of $\sigma_{\widetilde{Y}}^2$ in (14) to $\sigma_{\widehat{Y}}^2$ in (13) as a measure of the error introduced through the elimination of the crossband filters. This ratio is

$$\begin{aligned} \frac{\sigma_{\widetilde{Y}}^2}{\sigma_{\widehat{Y}}^2} &= \frac{\|\Gamma\theta^{(i)} - \widehat{\Gamma}\theta^{(i)}\|^2}{\|\theta^{(i)}\|^2 + \|\Gamma\theta^{(i)}\|^2} \\ &\leq \frac{\max_i \sigma_i(\Gamma - \widehat{\Gamma})}{1 + \min_i \sigma_i(\Gamma)} \end{aligned}$$

with $\sigma_i(\Gamma)$ denoting the i^{th} singular value of matrix Γ.

The bound on the relative approximation error power given above then lends itself to a quantitatively based selection process for which crossband filters to keep. A candidate procedure is as follows.

1. Select a desired bound α on $\sigma_{\widetilde{Y}}^2/\sigma_{\widehat{Y}}^2$. Set $i = 0$, and form $\widehat{\Gamma}_0 = 0$, corresponding to the use of no crossband filters.

2. Calculate

$$\frac{\max_i \sigma_i(\Gamma - \widehat{\Gamma}_i)}{1 + \min_i \sigma_i(\Gamma)}$$

and compare it to α. If it is less than α, stop, and use only the crossband filters reflected in $\widehat{\Gamma}_i$. If not, proceed to step 3.

3. Increment $i = i + 1$, and set $\widehat{\Gamma}_i$ to be Γ with all blocks corresponding to crossband filters between bands further than i bands apart to zero. In this way $\widehat{\Gamma}_i$ includes crossbands between bands j and k if $|j - k| \leq i$.

4. Go to step 2.

Once this procedure has ended, one can proceed further to address limiting the length of the crossband filters. For crossband filters between bands separated by a distance of i bands (the largest separation), one can compute the relative approximation error bound as the length of these bands is decreased to see if further simplifications are possible.

5 Implementation and the Adaptive Algorithm

To implement that CSAF, one needs to construct the subband outputs as generated through the inband and crossband filters. The implementation that we present below exploits the relationship of the crossband filter parameters to the inband filter parameters to express the subband outputs only in terms of the inband filter parameters. This output construction has the added advantage of making the algorithm specification straightforward, as well as imparting some computational efficiency to the overall implementation.

To proceed, note that given the definitions for the filter parameters $\mathbf{c}_{i,j}$ from (6) and the subband regressor $\phi_i(k)$ from (12), we have

$$\begin{aligned}
\widehat{Y}_i(k) &= \sum_{j=0}^{M-1} \phi_j(k)\mathbf{c}_{i,j}(k) \\
&= \phi_i(k)\mathbf{c}_{i,i}(k) + \sum_{j \neq i} \phi_j(k) \left(\sum_{\ell=0}^{M-1} \Gamma_{i,j}^{\ell} \mathbf{c}_{\ell,\ell}(k) \right) \\
&= \left(\phi_i(k) + \sum_{j \neq i} \phi_j(k) \Gamma_{i,j}^{i} \right) \mathbf{c}_{i,i}(k) + \sum_{\ell \neq i} \left(\sum_{j \neq i} \phi_j(k) \Gamma_{i,j}^{\ell} \right) \mathbf{c}_{\ell,\ell}(k).
\end{aligned}$$

If we define the modified regressors

$$X_{i,\ell}(k) = \begin{cases} \phi_i(k) + \sum_{j \neq i} \phi_j(k) \Gamma_{i,j}^{i}, & i = \ell \\ \sum_{j \neq i} \phi_j(k) \Gamma_{i,j}^{\ell}, & i \neq \ell \end{cases} \tag{15}$$

then we may express each $\widehat{Y}_i(k)$ as

$$\widehat{Y}_i(k) = \sum_{j=0}^{M-1} X_{i,j}(k)\mathbf{c}_{j,j}(k).$$

Note that in the construction of $X_{i,\ell}(k)$ in (15), one may use $\widehat{\Gamma}$ in place of Γ if any crossband filters have been eliminated in the constraints. Figure 2 shows the implementation.

There are two approaches to the specification of an adaptive algorithm. The first adjusts each set of parameters $\mathbf{c}_{i,i}(k)$ in order to minimize

$$J_{\text{total}} = \sum_{j=0}^{M-1} E_j^2(k).$$

This LMS algorithm to do this is

$$\mathbf{c}_{i,i}(k+1) = \mathbf{c}_{i,i}(k) + \mu \sum_{j=0}^{M-1} X_{j,i}(k) E_j(k)$$

for each $i = 0, \ldots, M-1$. This algorithm accounts for errors in all subbands to which errors in $\mathbf{c}_{j,j}$ contribute.

A simpler algorithm is to update $\mathbf{c}_{i,i}(k)$ based only upon the error in subband i; that is, to minimize $J_i(k) = E_i^2(k)$. The LMS algorithm for doing this is

$$\mathbf{c}_{i,i}(k+1) = \mathbf{c}_{i,i}(k) + \mu X_{i,i}(k) E_i(k)$$

for each $i = 0, \ldots, M-1$.

In [6], a convergence analysis for these algorithms is given that shows that the convergence of the inband filter parameters is approximately decoupled. Simulation examples demonstrate that the algorithms provide improved convergence speed relative to the unconstrained implementation of [1]. The computational complexity of the CSAF, if implemented while retaining only crossband filters between adjacent bands, is comparable to that of fullband filters, becoming less as N increases.

6 Conclusions

We have developed a constrained subband adaptive filter (CSAF) to minimize the aliasing effects. The constraint formulation also provides a procedure with which to evaluate quantitatively if any of the crossband filters may be neglected in the implementation. Two adaptation algorithms have been demonstrated for the CSAF. The algorithms display improved convergence speed with respect to unconstrained SAFs, and maintain a computational complexity comparable to that of fullband adaptive filters.

References

[1] A. Gilloire and M. Vetterli, "Adaptive filtering in subbands with critical sampling: analysis, experiments, and application to acoustic echo cancellation," *IEEE Trans. Signal Processing*, vol. 40, no. 8, pp. 1862-1875, August 1992.

[2] J.J. Shynk, "Frequency-domain and multirate adaptive filtering," *IEEE Signal Processing Mag.*, vol. 9, no. 1, pp. 14-37, January 1992.

[3] H. Yasukawa, S. Shimada, and I. Furukawa, "Acoustic echo canceller with high speech quality," in *Proc. ICASSP 87*, pp. 2125-2128, Dallas, TX, 1987.

[4] W. Kellermann, "Analysis and design of multirate systems for cancellation of acoustic echoes," in *Proc. ICASSP 88*, pp. 2570-2573, New York, NY, 1988.

[5] P.P. Vaidyanathan, *Multirate Systems and Filter Banks*. Englewood Cliffs, NJ: Prentice-Hall, 1993.

[6] S. Zimmermann, *Subband Vector Space Adaptive Filtering*. Ph.D. Thesis: Illinois Institute of Technology, May 1996.

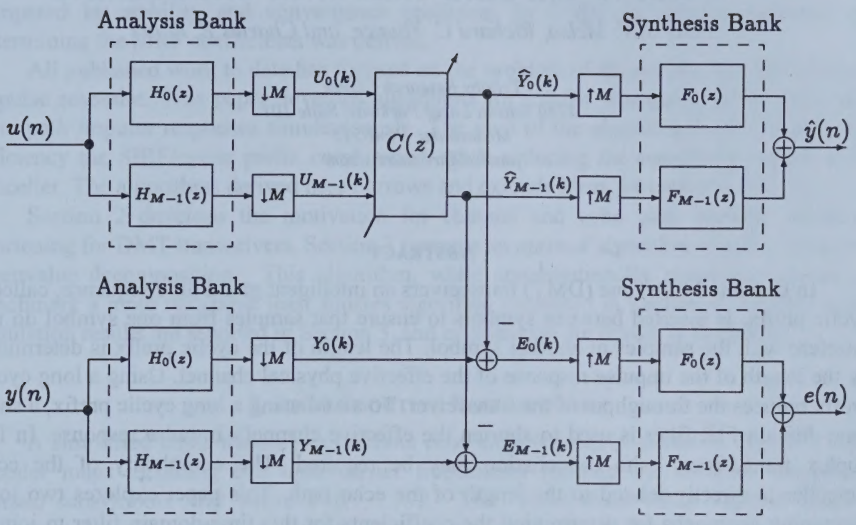


Figure 1: N-channel subband adaptive filter.

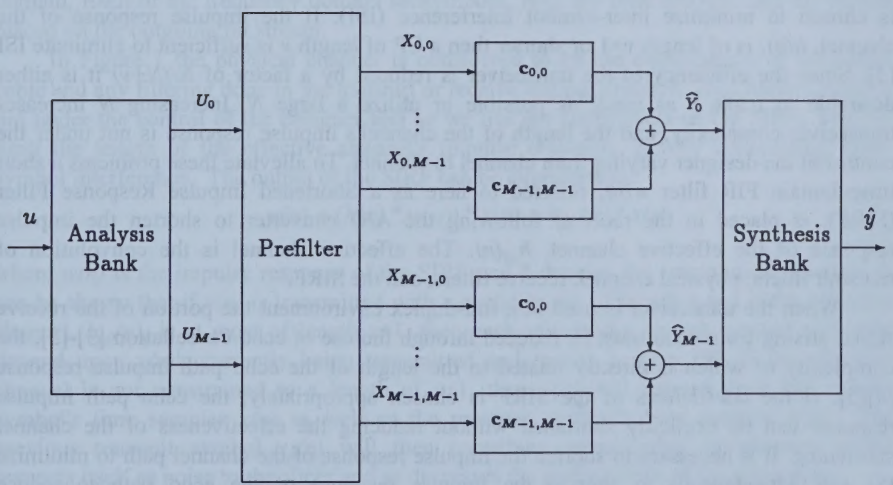


Figure 2: CSAF implemenation.

JOINT IMPULSE RESPONSE SHORTENING

Peter J.W. Melsa, Richard C. Younce, and Charles E. Rohrs

Tellabs Research Center
3740 Edison Lakes Parkway, Suite 101
Mishawaka, IN 46545
pmelsa@trc.tellabs.com

ABSTRACT

In Discrete MultiTone (DMT) transceivers an intelligent guard time sequence, called a cyclic prefix, is inserted between symbols to ensure that samples from one symbol do not interfere with the samples of another symbol. The length of the cyclic prefix is determined by the length of the impulse response of the effective physical channel. Using a long cyclic prefix reduces the throughput of the transceiver. To avoid using a long cyclic prefix, a short time-domain FIR filter is used to shorten the effective channel's impulse response. In full duplex transceivers echo cancellation may be required. The complexity of the echo canceller is directly related to the length of the echo path. This paper explores two joint shortening approachs for determining the coefficients for this time-domain filter to jointly shorten both the channel and echo path impulse responses.

1. INTRODUCTION

In Discrete MultiTone (DMT) transceivers each symbol is comprised of N samples to be transmitted to the remote receiver plus a cyclic prefix (CP) of length v [1]. The CP length is chosen to minimize inter-symbol interference (ISI). If the impulse response of the channel, $h(n)$, is of length $v+1$ or shorter then a CP of length v is sufficient to eliminate ISI [2]. Since the efficiency of the transceiver is reduced by a factor of $N/(N+v)$ it is either desirable to make v as small as possible or utilize a large N. Increasing N increases transceiver complexity, and the length of the channel's impulse response is not under the control of the designer varying from channel to channel. To alleviate these problems a short time-domain FIR filter $w(n)$, referred to here as a Shortened Impulse Response Filter (SIRF), is placed in the receiver following the A/D converter to shorten the impulse response of the effective channel, $h_{eff}(n)$. The effective channel is the convolution of transmit filters, physical channel, receive filters, and the SIRF.

When the transceiver is used in a full-duplex environment the portion of the receive signal arising from echo must be reduced through the use of echo cancellation [3]-[5], the complexity of which is directly related to the length of the echo path impulse response [4][5]. If the coefficients of the SIRF is chosen appropriately, the echo path impulse response can be explicitly shortened without reducing the effectiveness of the channel shortening. It is necessary to shorten the impulse response of the channel path to minimize ISI and advantageous to shorten the impulse response of the echo path to reduce complexity.

This paper addresses the calculation of the coefficients for the Shortened Impulse Response Filter (SIRF). It was shown in [6] that the greatest throughput, for a given computational complexity, of a DMT transceiver can be accomplished using an equalizer that shortens the impulse response of the channel and a cyclic prefix. This approach was

demonstrated for a DMT based HDSL transceiver in [1]. In [7] an efficient means for determining the coefficients of the SIRF was developed. However, the algorithms in [7] are hampered by stability and convergence problems. In [2][8] an optimal approach to determining the SIRF coefficients was derived.

All published work to date has focused on the problem of shortening only the channel impulse response. This paper develops algorithms for jointly shortening the channel and echo path impulse responses simultaneously. The goal of the algorithm is to maintain the efficiency the SIRF/cyclic prefix combination while reducing the complexity of the echo canceller. The algorithms derived here borrows and extends ideas presented in [8]-[10].

Section 2 develops the motivation for channel and echo path impulse response shortening for DMT transceivers. Section 3 presents an *optimal* algorithm which is based on eigenvalue decomposition. This algorithm, while computationally expensive, serves to benchmark a more realizable least squares algorithm presented in Section 4. Finally, some simulation results are reported in Section 5, and conclusions are drawn in Section 6.

2. Joint Shorting Background

A simplified DMT transceiver (without echo cancellation) is shown in Figure 1. The encoder maps incoming bits onto carrier frequencies forming the complex frequency domain subsymbols $X(k)$ where $k=0,1,....N/2$. The subsymbols taken together form a frequency domain symbol. A conjugate symmetry condition is then imposed creating a length N frequency domain signal. This signal is inverse Fourier transformed to generate a real, length N time domain symbol, $x(n)$, which is prefixed with a Cyclic Prefix (CP) of length v and transmitted to the far end receiver. The CP consists of the last v samples of $x(n)$. At the receiver, the signal is passed through a short time-domain FIR filter, and then the CP is removed. This signal, $y(n)$ is then Fourier transformed back to the frequency domain. Each of the frequency domain subsymbols, $Y(k)$, are then decoded and the results placed on the outgoing bit stream.

In Figure 1 the physical channel is considered to be the combination of the actual cable and any filtering done in the transmit or receive analog circuitry. The channel, $h(n)$, is not under the control of the designer and so we use the coefficients of the SIRF, $w(n)$, to limit the length of the effective channel's impulse response, thereby minimizing inter-symbol interference. The output of the SIRF can be expressed as,

$$y(n) = (h(n) * w(n)) * x(n) = h_{eff}(n) * x(n) \tag{1}$$

where $w(n)$ is the impulse response of the SIRF and * denotes the convolution operator. It can be shown that if $x(n)$ is transmitted with a cyclic prefix of length v and if the effective channel, $h_{eff}(n)$, is at most of length $v+1$, then each $y(n)$ in the current symbol will only depend upon $x(n)$'s currently being transmitted and results in zero ISI. If the effective channel is not constrained to a length of $v+1$, then $y(n)$ will depend upon the current symbol's time samples $x(n)$ as well as the previous symbol's time samples $x_p(n)$. The previous transmit symbol $x_p(n)$ will then contribute inter-symbol interference which presents itself as noise to the slicer and so decreases the performance of the transceiver.

Regardless of the choice of $w(n)$ it is generally not possible to shorten the impulse response perfectly. Some energy will lie outside the largest $v+1$ consecutive samples of $h_{eff}(n)$. As a measure of the inter-symbol interference we can measure the shortening SNR (SSNR) which is the ratio of the energy in the largest consecutive $v+1$ samples to the energy in the remaining samples. The largest $v+1$ samples will not necessarily start with the first

sample. This delay, d, is normally compensated for at the receiver by delaying the start of the receive symbol. The shape of the resulting impulse response of the effective channel, $h_{eff}(n)$, is usually unimportant, what is important is that the SSNR be maximized.

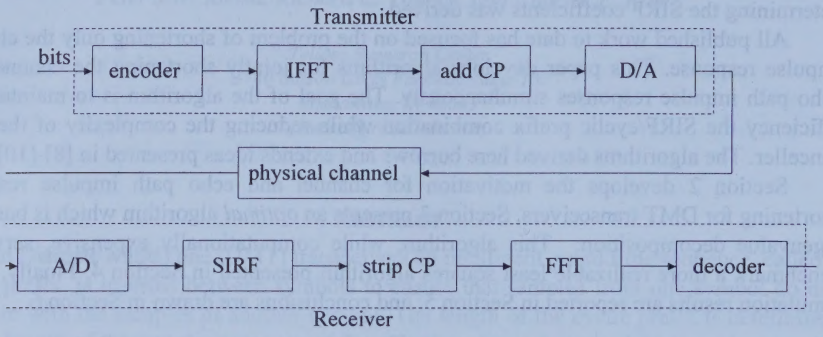


Figure 1. Simplified DMT Transmitter/Receiver

When a DMT transceiver is used for bi-directional communication over a single cable there are two signal paths to consider: channel and echo. The channel is associated with communication between two remote transceivers and carries the actual data from one end to the other end. An echo path also exists between a local transmitter and receiver due to an impedance mismatch in the hybrid circuit. The receive signal is the combination of the far-end signal through the physical channel and the near-end signal through the echo path. The motivation for impulse response shortening provided above related to the channel path. Typically, echo canceller structures for DMT transceivers combine both frequency domain and time domain components. The complexity of the time domain portion is directly related to the length of the echo path impulse response [4][5]. Thus, impulse response shortening of the echo path is desirable to reduce computational complexity in the echo canceller. The reader is referred to [3]-[5] for discussions of echo canceller structures for DMT transceivers. The important concept for the discussion here is that it is necessary to shorten the impulse response of the channel path, and advantageous to shorten the impulse response of the echo path to reduce transceiver complexity.

A necessary step in computing SIRF coefficients is to estimate the impulse responses. Typically, DMT transceivers use known training sequences for purposes of adapting various parameters associated with the transmitter and receiver. Since DMT transceivers have the frequency response naturally available, the training sequences can be used to calculate the frequency responses of the channel and echo path. The frequency responses can then be inverse transformed into the time domain, generating the time-domain impulse responses. Care should be taken to reduce the variance of the estimation noise, and compensate for any other known effects such as aliasing.

3. JOINT OPTIMAL ALGORITHM

We can expressed the convolution operation in Eq.(1) in matrix form as $h_{eff}(n) = Hw$ where w is a tx1 column vector containing the SIRF coefficients and H is composed of samples of the impulse response shifted so that Hw results in the convolution of h with w. Similarly, the effective echo impulse response can be written as $g_{eff}=Gw$. Let M be the length of $h(n)$, h_{win} represent a window of v+1 consecutive samples of h_{eff} starting

at sample d, and let h_{wall} represent the remaining samples of h_{eff}. Additionally, let g_{win} represent a window of v_e+1 consecutive samples of g_{eff} starting at sample d_e , and let g_{wall} represent the remaining samples of g_{eff}. These definitions can be expressed as

$$h_{win} = H_{win}w, \quad h_{wall} = H_{wall}w,$$
$$g_{win} = G_{win}w, \quad \text{and} \quad g_{wall} = G_{wall}w,$$

where H_{win}, H_{wall}, G_{win}, and G_{wall} are formed by taking appropriate rows of H and G. For example,

$$H_{win} = \begin{bmatrix} h(d) & h(d-1) & \cdots & h(d-t+1) \\ h(d+1) & h(d) & \ldots & h(d-t+2) \\ \vdots & & \ddots & \vdots \\ h(d+v) & h(d+v-1) & \cdots & h(d+v-t+1) \end{bmatrix}$$

One way to formulate joint optimal shortening is to choose w to minimize the energy outside the windows

$$w^T\left(\alpha H_{wall}^T H_{wall} + (1-\alpha)G_{wall}^T G_{wall}\right)w$$
$$= w^T A(\alpha)w$$

such that the energy inside the windows is constrained.

$$w^T\left(\beta H_{win}^T H_{win} + (1-\beta)G_{win}^T G_{win}\right)w$$
$$= w^T B(\beta)w=1$$

The constraint ensures that the trivial solution, $w = [0 \ \ldots \ 0]^T$ is disallowed. The parameters α and β balance the energy relationship between the energy in and out of the window for both the channel and echo. A full development of the solution to this problem can be found in [2]. The matrix $B(\beta)$ is always symmetric positive semi-definite. If it is further assumed that $B(\beta)$ is actually positive definite the optimal SIRF coefficients are given by

$$w_{opt} = \left(\sqrt{B(\beta)}^T\right)^{-1} l_{\min},$$

where $l_{\min}$ is the unit-length eigenvector corresponding to the minimum eigenvalue $\lambda_{\min}$ of C. The matrix C is expressed as

$$C = \left(Q\sqrt{\Lambda}\right)^{-1} A\left(\sqrt{\Lambda}Q^T\right)^{-1},$$

where Λ is a diagonal matrix formed from the eigenvalues of $B(\beta)$, and the columns of Q are the orthonormal eigenvectors.

This algorithm results in the optimally shortened convex sum of the effective channel and echo impulse responses. It provides superior performance over the least squares method presented in the next section but at a high computational costs making it impractical for real time systems.

4. Least Squares Joint Shortening

This joint shortening approach seeks to fit a pole-zero-zero model (see Figure 2) to the echo and the channel impulse responses. Assume that the original impulse of the channel is represented as a transfer function $h(z^{-1}) = a(z^{-1}) / (1 + b(z^{-1}))$ and the echo as $h_e(z^{-1}) = c(z^{-1}) / (1 + b(z^{-1}))$. The joint least-squares approach finds the best pole-zero-zero model with transfer function $\hat{a}(z^{-1}) / (1 + \hat{b}(z^{-1}))$ for the channel and $\hat{c}(z^{-1}) / (1 + \hat{b}(z^{-1}))$

for the echo. If the poles of the model, $1+\hat{b}(z^{-1})$, are used for the coefficients of the SIRF, then the effective channel will have a transfer function of $\hat{a}(z^{-1})$ and the effective echo will have a transfer function of $\hat{c}(z^{-1})$. The SIRF cancels the poles of both the physical channel and echo path, leaving the zeros of the model. If the number of zeros in the model for the channel is chosen to be v+1 then the resulting effective channel impulse response will be approximately of length v+1. The number of zeros in the model for the echo is chosen based upon the desired computational complexity of the echo canceller. It is worth noting that one can not make the echo response arbitrarily short with a limited number of poles in the model without adversely affecting the SSNR of the effective channel response.

Throughout the remainder of this paper it is assumed that *z(n)* is zero-mean, white and independent of *x(n)*, and that the echo is stationary and causal.

Since the channel response must be shortened to remove inter-symbol interference, while echo shortening is beneficial to reduce echo canceller complexity, the number of zeros used for the two responses do not have to be equal. For simplicity of derivation, we assume that the number of desired zeros modeled for both the channel and echo is v+1. It is also assumed that the length of the SIRF, t, is greater than or equal to v. It is straight forward to generalize the algorithm to a more generic case. The following derivation also places all the poles and zeros starting at the beginning of the impulse responses. Potentially, modeling the impulse responses from the beginning wastes some of the zeros available on flat delay or small, insignificant samples of the impulse response. To avoid this problem, we can adjust the location of the zeros to take advantage of the flat delay of the channel.

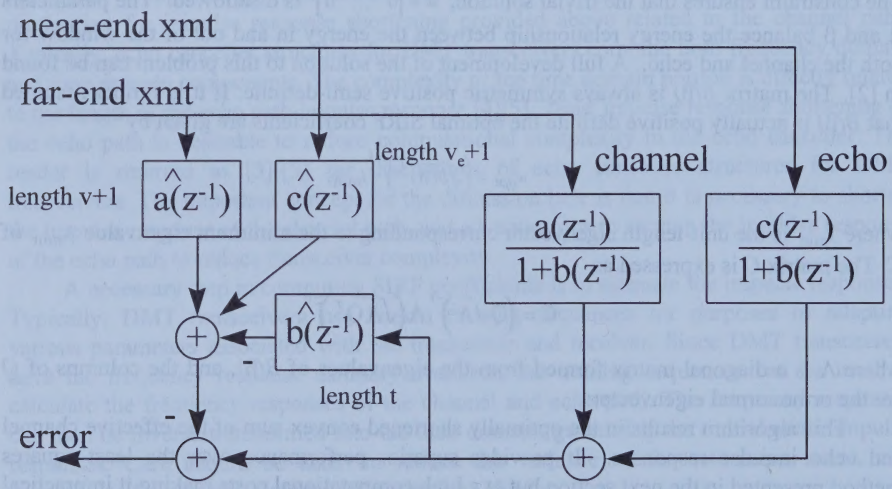


Figure 2. Pole-Zero-Zero Model for Joint Shortening

The joint shortening problem can be solved using a standard least-squares approach. Since such an approach requires the inversion of a large matrix it is infeasible for real-time system implementation. To avoid these problems we will embed the pole-zero-zero model into a multi-channel AR model for which efficient solutions exist [9][10].

The pole-zero-zero model can be rewritten as

$$\hat{y}_k = -\hat{b}_1 y_{k-1} - \ldots - \hat{b}_t y_{k-t} + \hat{a}_0 x_k + \ldots + \hat{a}_v x_{k-v} + \hat{c}_0 z_k + \ldots + \hat{c}_v z_{k-v}.$$

As is typical with AR modeling, the algorithm computes the t+2v+2 parameters recursively. In the j^{th} recursion ($1 \le j \le t$) of the algorithm the (j,j-δ,j-δ) pole-zero-zero model is generated from the (j-1,j-δ-1,j-δ-1) pole-zero-zero model, where $\delta = t - v \ge 0$. The (j,j-δ,j-δ) pole-zero-zero model can be written as

$$y_k = -\hat{b}_1^j y_{k-1} - \ldots - \hat{b}_j^j y_{k-j} + \hat{a}_0^j x_k + \ldots + \hat{a}_{j-\delta}^j x_{k-j+\delta} + \hat{c}_0^j z_k + \ldots + \hat{c}_{j-\delta}^j z_{k-j+\delta} + e_k^j \,. \tag{2}$$

Defining $u_k = \left[y_k \; x_{k+\delta} \; z_{k+\delta} \right]^T$, Eq. (2) can be rewritten into the from of a multi-channel AR model

$$u_k = \begin{bmatrix} -\hat{b}_1^j & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{k-1} \\ x_{k-1+\delta} \\ z_{k-1+\delta} \end{bmatrix} + \ldots + \begin{bmatrix} -\hat{b}_\delta^j & \hat{a}_0^j & \hat{c}_0^j \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{k-\delta} \\ x_k \\ z_k \end{bmatrix} + \ldots$$

$$+ \begin{bmatrix} -\hat{b}_{\delta+1}^j & \hat{a}_1^j & \hat{c}_1^j \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{k-\delta-1} \\ x_{k-1} \\ z_{k-1} \end{bmatrix} + \ldots + \begin{bmatrix} -\hat{b}_j^j & \hat{a}_{j-\delta}^j : \hat{c}_{j-\delta}^j \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} y_{k-j} \\ x_{k-j+\delta} \\ z_{k-j+\delta} \end{bmatrix} + \begin{bmatrix} e_k^j \\ x_{k+\delta} \\ z_{k+\delta} \end{bmatrix}$$

$$u_k = -\Theta_1^j u_{k-1} - \ldots - \Theta_j^j u_{k-j} + \begin{bmatrix} e_k^j \\ x_{k+\delta} \\ z_{k+\delta} \end{bmatrix} \tag{3}$$

where

$$-\Theta_i^j = \begin{cases} \begin{bmatrix} -\hat{b}_j^j & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & 1 \le i \le \delta - 1 \\ \begin{bmatrix} -\hat{b}_i^j & \hat{a}_{i-\delta}^j & \hat{c}_{i-\delta}^j \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} & \delta \le i \le j \end{cases}$$

Multiplying both sides of Eq. (3) by u_{k-i}^T and taking expectations and noting that the error, e_k^j is orthogonal to $x_{k+\delta}^j$ and $z_{k+\delta}^j$ at each iteration yields

$$R(0) + \Theta_1^j R(-1) + \ldots + \Theta_j^j R(-j) = \Sigma_j^f \quad \text{for } i = 0 \tag{4}$$

$$R(i) + \Theta_1^j R(i-1) + \ldots + \Theta_j^j R(i-j) = 0 \quad \text{for } 1 \le i \le j \tag{5}$$

where

$$R(i) = E\left\{ u_k u_{k-i}^T \right\} = \begin{bmatrix} R_{yy}(i) & R_{yx}(i-\delta) & R_{yz}(i-\delta) \\ R_{xy}(i+\delta) & R_{xx}(i) & 0 \\ R_{zy}(i+\delta) & 0 & R_{zz}(i) \end{bmatrix} \tag{6}$$

$$R(-i) = R^T(i), \tag{7}$$

and

$$\Sigma_j^f = \begin{bmatrix} R_{ee}(0) & 0 & 0 \\ R_{xy}(\delta) & R_{xx}(0) & 0 \\ R_{zy}(\delta) & 0 & R_{zz}(0) \end{bmatrix}. \tag{8}$$

Assuming that

$$y(n) = h(n) * x(n) + h_e(n) * z(n)$$

then

$$R_{xx}(k) = S_x \delta(k)$$

$$R_{zz}(k) = S_z \delta(k)$$

$$R_{xy}(k) = R_{yx}(-k) = S_x h(-k)$$

$$R_{zy}(k) = R_{yz}(-k) = S_z h_e(-k)$$

and

$$R_{yy}(k) = S_x \sum_{i=0}^{\infty} h(i)h(i-k) + S_z \sum_{i=0}^{\infty} h_e(k)h_e(i-k)$$

where S_x and S_z are the power in $x(n)$ and $z(n)$ respectively.

Eqs. (4)-(8) represent a j^{th} order multi-channel AR model. This system of linear equations can be solved efficiently using the multi-channel version of the Levinson Algorithm [10][11]. After the $j=t$ recursion, the (t,v,v) pole-zero-zero model coefficients can then be read from Θ'_i, $1 \le i \le t$.

The advantage of this algorithm over a least-squares approach is that computing the pole-zero-zero model only requires the inversion of 3x3 matrices and 3x3 matrix operations for t recursions, where a least squares approach requires one recursion with the inversion of an $(t+2v+2)$x$(t+2v+2)$ matrix.

5. Simulation Results

To demonstrate the capabilities of the joint shortening algorithm presented above the configuration shown in Figure 3 is used. The physical cable used is 9 kft of 26 gauge unshielded twisted pair, and the bandwidth used is 20-900 kHz. The echo path is a transmit path of 20-138 kHz through a hybrid. This configuration is representative of that encountered in the central office (ATU-C) to remote terminal (ATU-RT) path in the Asymmetrical Digital Subscriber Lines (ADSL) environment [12]. The impulse responses for this configuration were provided for us by W.Y. Chen of Bellcore [13]. For the simulation we used a SIRF length of 16 (t=16), a cyclic prefix of length 32 (v=32), a target effective echo response length of 33 (v_e=32), and $\alpha = \beta = .5$.

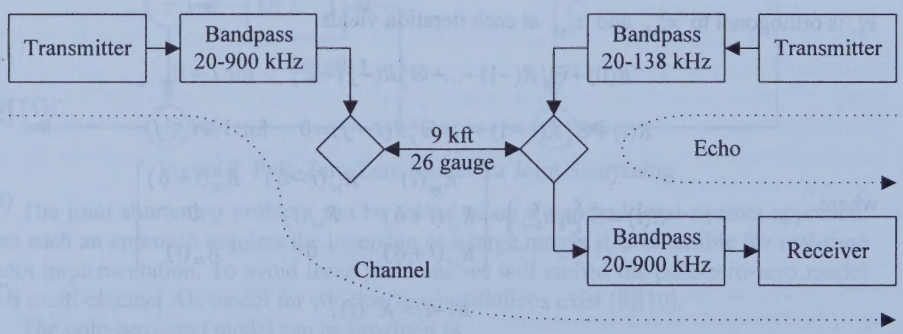


Figure 3. Configuration for Simulation Example

Figure 4 shows the original impulse response of the channel and the effective shortened channel if the optimal joint shortening algorithm is used. A SSNR=56.2 dB was achieved. Figure 5 shows the original echo impulse response and the effective shortened echo for a channel only optimal shorting algorithm [8] and the least squares joint shortening presented in this paper. As shown, both have shortened the echo response, however, the least squares joint shortening algorithm which explicitly works to shorten the echo has done a significantly better job. The optimal joint shortening results are not shown because they are indistinguishable from the least squares results.

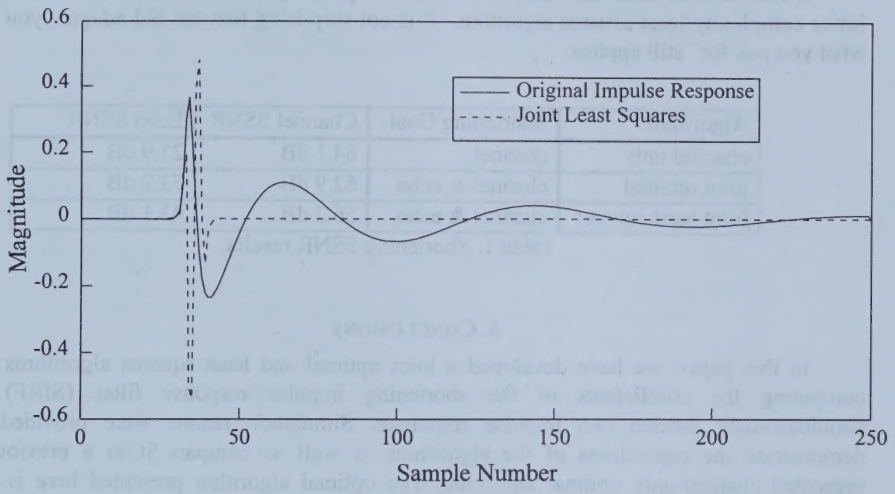


Figure 4. Original Channel Response & Jointly Shortened Effective Response

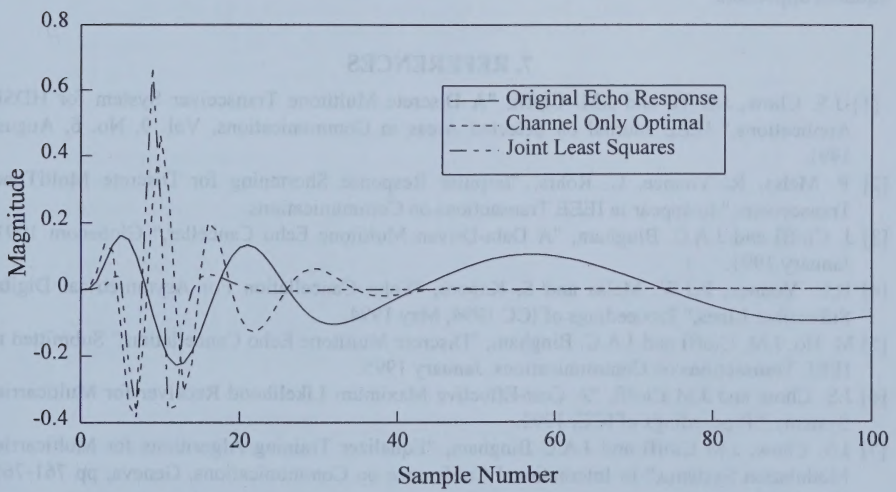


Figure 5. Original Echo Impulse Response and Effective Response Resulting from Optimal and channel only Joint Shortening

Table 1 gives the resulting channel and echo SSNR values for the channel only shortening and both joint algorithms. The channel only algorithm achieves its goal of optimizing the *channel* shortening to a level of 64 dB SSNR. It outscores the joint algorithms in this goal. However, the channel only algorithm only achieves 23 dB of echo impulse response shorting. The joint algorithm beats this by 20 to 30 dB while only loosing 2 to 6 dB of channel shorting. The small loss of SSNR is insignificant in practical transceiver design but the large gains in echo shortening represents significant reduction in echo canceller complexity. Herein lies the chief advantage of the joint shortening method.

It can be seen from the table that the joint optimal produces better results than the lower complexity least squares algorithm. It is not surprising that the old adage: "you get what you pay for" still applies.

Algorithm	Shortening Goal	Channel SSNR	Echo SSNR
channel only	channel	64.1 dB	23.9 dB
joint optimal	channel & echo	62.9 dB	53.2 dB
joint least squares	channel & echo	56.2 dB	43.1 dB

Table 1. Shortening SSNR results

6. Conclusions

In this paper, we have developed a joint optimal and least squares algorithms for computing the coefficients of the shortening impulse response filter (SIRF) to simultaneously shorten two impulse responses. Simulation results were provided to demonstrate the capabilities of the algorithms as well as compare it to a previously presented channel-only optimal algorithm. The optimal algorithm presented here is not practical for real-time implementation but serves to benchmark a more practical least squares approach.

7. REFERENCES

[1] J.S. Chow, J.C Tu and J.M. Cioffi, "A Discrete Multitone Transceiver System for HDSL Applications," IEEE Journal on Selected Areas in Communications, Vol. 9, No. 6, August 1991.

[2] P. Melsa, R. Younce, C. Rohrs, "Impulse Response Shortening for Discrete MultiTone Transceivers," to appear in IEEE Transactions on Communications.

[3] J. Cioffi and J.A.C. Bingham, "A Data-Driven Multitone Echo Canceller," Globecom 1991, January 1991.

[4] R.C. Younce, P.J.W. Melsa and S. Kapoor, "Echo Cancellation For Asymmetrical Digital Subscriber Lines," Proceedings of ICC 1994, May 1994.

[5] M. Ho, J.M. Cioffi and J.A.C. Bingham, "Discrete Multitone Echo Cancellation," Submitted to IEEE Transactions on Communications, January 1995.

[6] J.S. Chow and J.M Cioffi, "A Cost-Effective Maximum Likelihood Receiver for Multicarrier Systems," Proceedings of ICC, 1992.

[7] J.S. Chow, J.M Cioffi and J.A.C Bingham, "Equalizer Training Algorithms for Multicarrier Modulation Systems," In International Conference on Communications, Geneva, pp 761-765, May 1993.

[8] P.J.W. Melsa, R.C. Younce, C.E. Rohrs, "Optimal Impulse Response Shortening," 33rd Annual Allerton Conference on Communication, Control, and Computing, October 1995.

[9] D.T. Lee, B. Friedlander and M. Morf, "Recursive Ladder Algorithms for ARMA Modeling," IEEE Transactions on Automatic Control, Vol. AC-27, No. 4, August 1982.
[10] N. Al-Dhahir, A. Sayed and J.M. Cioffi, "A High-Performance Cost-Effective Pole-Zero MMSE-DFE," In Proceedings Allerton Conference on Communication, Control and Computing, September 1993.
[11] S. Kay, Modern Spectral Estimation: Theory and Applications, Prentice Hall, 1988.
[12] T1E1.4/95-007 Draft Asymmetric Digital Subscriber Line (ADSL) Metallic Interface Specification, Committee T1-Telecommunications, January 1995.
[13] W.Y. Chen, Bellcore, private communication, July 1993.

NONLINEAR, HIGH-DIMENSIONAL ADAPTIVE SIGNAL PROCESSING WITH AN APPLICATION TO ACTIVE NOISE CONTROL

DR. RICHARD H. BURKHART

ABSTRACT. Adaptive methods, such as the Widrow-Hoff LMS method, provide simple, robust, and flexible approaches to many problems in digital signal processing and control. However these methods generally require considerable fine-tuning and have usually been limited to linear, low-dimensional systems, predominately single-input single-output. We describe a more sophisticated, yet still practical, approach to high dimensional digital control where the adaptive coefficients require a nonlinear estimation algorithm. In our application to broad band 3D active noise control, the coefficients are those of IIR filters for identification and for feedforward adaptive control. Stability of the IIR filters is guaranteed by the Shur-Cohn algorithm, and estimation is via our iterative Newton-GMRES solver. Real time operation was achieved with tens of input and output data streams and hundreds of coefficients.

1. INTRODUCTION

Adaptive methods, such as the Widrow-Hoff LMS method [Wi], provide simple, robust, and flexible approaches to many problems in digital signal processing and control. However these methods generally require considerable fine-tuning and have usually been limited to linear, low-dimensional systems, predominately single-input single-output. We describe a more sophisticated, yet still practical, approach to high dimensional digital control where the adaptive coefficients require a nonlinear

Date. October 3, 1996.

1991 *Mathematics Subject Classification.* Primary 93C40, 93C10, 93C57.

Key words and phrases. adaptive control, active noise control, nonlinear, Newton, GMRES, Kalman filter.

Thanks to my collaborators, Andy Anderson and Rich Russo, of Boeing Noise Research.

estimation algorithm. Real time operation was achieved with tens of input and output data streams and hundreds of coefficients.

In our application to broad band 3D active noise control [Ne], the coefficients are those of IIR filters for identification and for feedforward adaptive control, optionally combined with feedback control. The recursive update process is formulated as a modified Kalman filter. Stability of the IIR filters is guaranteed by the Shur-Cohn algorithm, or Jury's test [El, p. 293]. Estimation is via a nonlinear version of the iterative GMRES solver [Ke], sometimes referred to as an inexact Newton method or Newton-GMRES. We use a Levenburg-Marquardt trajectory search for damping, i.e., to extend the region of convergence [Fl]. In its simplest form, this solver is equivalent to a gradient, or normalized LMS, method with automatic estimation of the optimal scaling coefficient. With more computation it yields the full nonlinear least squares solution. With an intermediate amount of computation it will give a solution that is part way between a gradient and a full least squares update, which often means faster convergence of the coefficients. The engineer can easily vary the key parameter, the number of search directions, or Krylov subspace dimension, to determine the best trade-offs.

The linear version of GMRES may be applied to linear estimation, such as the more traditional adaptation of FIR coefficients. In both the linear and nonlinear cases a batch approach is taken to the updating operation. That is, the streams of sampled input and output data are not processed one-by-one but are recorded over a fraction of a second. Then this batch of data is processed as a single high-dimensional problem to update the coefficients. Meanwhile the control filters are running continuously in the background, using a double buffer to switch instantly to the new set of coefficients after the update.

In our feedforward control application to noise control in a 3D enclosure, there is a system of source sensors, such as microphones, to measure the incoming noise as close to the physical sources as possible. Then filters are applied to these signals to drive a system of actuators, such as speakers. Finally, there is a system of control, or feedback, sensors distributed in the desired quiet zone. The response of the feedback sensors to the actuators is used to adapt the filter coefficients so that the actuator signals sum to cancel the physical source noise as closely as possible in the target region.

This scheme requires the additional estimation of the actuator-to-feedback sensor prediction function, also represented by a filter. This identification step is done off-line, prior to control, with possible on-line

updates alternating with control updates. However, on-line identification requires the addition of broad band noise to the actuator signals, degrading control performance.

For some active noise control applications FIR filters are quite adequate. But if there is significant reverberation in the enclosure, then very long FIR filters might be required for the actuator-to-feedback prediction functions. IIR filters will sum the reverberations using many fewer coefficients. For control, IIR filters can be used to cancel any effects of acoustic feedback from the actuators to the source sensors.

2. The Prediction and Control Filters

For the IIR prediction filters we use the notation $c = G.u$, or

$$c(i) = \sum_{j=d_F}^{m_F} G_F(j)\, u(i-j) - \sum_{j=d_I}^{m_I} G_I(j)\, c(i-j) \tag{2.1}$$

where u = actuator signal, c = predicted signal, G_F = FIR coefficients, and G_I = IIR coefficients. The identification problem is to estimate $G = (G_F, G_I)$.

For IIR control filters, the notation is similar: $u = X.y$, or

$$u(i) = \sum_{j=e_F}^{n_F} X_F(j)\, y(i-j) - \sum_{j=e_I}^{n_I} X_I(j)\, u(i-j) \tag{2.2}$$

where y = sampled source input signal, X_F = FIR coefficients, and X_I = IIR coefficients.

Let p = the nuisance noise that we want to cancel, and

$$z = G.X.y + p \tag{2.3}$$

be the measured output (feedback signal). This assumes that G is an accurate representation of the prediction function. Then the goal of active noise control is to continually adjust $X = (X_F, X_I)$ to achieve cancellation; that is, $z \sim 0$.

3. The Modified Kalman Filter

The update equations are derived by minimizing a functional. For identification this functional is

(3.1)

$$F(G) = \frac{1}{2}\left(|G.u - z|^2 / D_z + |G - G_0|^2 / D_q + |G|^2 / D_p\right)$$

where u = actuator signal, z = measured data, G_0 = prior coefficient values, G = updated coefficient values to be estimated, and D_z, D_q, D_p = error variances. Here $G = G_{ij}$ is indexed over actuators

i and feedback sensors j, so that the vector $G.u$ represents the multi-dimensional IIR filter $\sum_i G_{ij}.u_i$ defined in terms of the one-dimensional IIR filters (2.1). Here it is assumed that filters G_{ij} and G_{lj} have the same IIR coefficients. That is, equation (2.1) becomes

(3.2)
$$c(i,k) = \sum_{l=1,}^{n_s} \sum_{j=d_F}^{m_F} G_F(j,l,k)\, u(i-j,l) - \sum_{j=d_I}^{m_I} G_I(j)\, c(i-j,k)$$

for $k = 1, ..., n_c$. Similarly equation (2.2) becomes

(3.3)
$$u(i,l) = \sum_{m=1,}^{n_e} \sum_{j=e_F}^{n_F} X_F(j,m,l)\, y(i-j,m) - \sum_{j=e_I}^{n_I} X_I(j,l)\, u(i-j,l)$$

for $l = 1, ..., n_s$.

Here the first two terms in (3.1) define an ordinary Kalman filter, except that D_z, D_q represent diagonal instead of full covariance matrices (for computational efficiency). Also, the IIR filters are nonlinear functions of the unknown coefficients G (though linear functions of u), making this a kind of extended Kalman filter. The third term represents an optional bias of G toward zero. In certain cases a small third term may aid stability, though in our application it did not appear to be needed. The point here is that the method may accommodate whatever additional terms in the functional that may prove useful for the application at hand.

The control functional is similar:

(3.4)
$$F(X) = \frac{1}{2}\left(|G.X.y - t|^2 / D_z + |X - X_0|^2 / D_q + |X|^2 / D_p \right)$$

where the target signal is $t = G.u - z$ for actuator signal $u = X_0.y$ and feedback sensor signal z. Note that $-t \sim p$ by equation (2.3).

In addition, we may accommodate feedback control by dividing the input signal y into two parts: $y = (y_s, y_f)$ for y_s = the source data (feedforward control) and $y_f = z - t$ = control data (feedback control). The latter is normally based only on the measured data z, but if z is close to zero and mostly white, which is the case after convergence, then it cannot provide effective control. On the other hand, the estimate $-t$ of the nuisance noise p, is the ideal basis for feedback since it is exactly what we want to control. However, we include both z and $-t$ because initially these roles are reversed, with $-t$ being a poor estimate of p while z is a good estimate.

Taking partials with respect to the coefficients of G, the functional (3.1) yields the nonlinear state update equations

$$(G.u)_G^t(G.u - z)/D_z + (G - G_0)/D_q + G/D_p = 0 \tag{3.5}$$

It is these equations that are solved to get G_{up} to a desired accuracy by GMRES. The corresponding updated covariance would normally be

$$\left((G.u)_G^t(G.u)_G/D_z + 1/D_q + 1/D_p\right)^{-1}$$

However this is non-diagonal, so only the diagonal of the first term is computed, yielding a major savings in computation:

$$(D_q)_{up} = \left(diag[(G.u)_G^t(G.u)_G]/D_z + 1/D_q + 1/D_p\right)^{-1} \tag{3.6}$$

The Kalman filter propagation uses the identity dynamics matrix and a diagonal process noise (gain) matrix. Thus $G = G_{prop} = G_{up}$ and $D_q = (D_q)_{prop} = (D_q)_{up} + D_w$ for process noise D_w which is tuned by the user. Our application only responds to change in D_w by orders of magnitude since the number of time samples in the batch of data is the primary determinant of gain, or weight on the data. That is, the greater the number of time samples, the greater the gain, as is seen by the dominance of the first term in equation (3.5) as the length of z increases. At the same time, a full covariance matrix is less necessary since much of the correlation is already present in the time history included in y, u, and z. Smoothing may be increased by overlapping batches of data, as well as by using a small prior variance D_q.

The partial derivatives $(G.u)_G$ are computed as follows. First compute $(G.u)(i) = c(i)$ for $i = t - m_t + 1, ..., t$ in formula (2.1) using $u(i)$ for $i = t - k_t + 1, ..., t$ where $k_t = m_t + m_F - 1$. Use the prior estimate of $c(i)$ for $i \leq t - m_t$ on the right hand side. Then the FIR partials are computed recursively for $n = d_F, ..., m_F$ by

$$c_{G_F(n,p,k)}(i,k) = u(i-n,p) - \sum_{j=d_I}^{M_i} G_I(j,k)\, c_{G_F(n,p,k)}(i-j,k) \tag{3.7}$$

where $M_i = \min(m_I, i-1)$. Similarly, the IIR partials are

$$c_{G_I(n,k)}(i,k) = -c(i-n,k) - \sum_{j=d_I}^{M_i} G_I(j)\, c_{G_I(n,k)}(i-j,k) \tag{3.8}$$

The propagation and update equations for the control Kalman filter are similar. Just note that $(G.X.y)_X = G.(X.y)_X$ where $(X.y)_X$ is

computed for values of $i = t - k_t + 1, ..., t$. This requires source time samples $y(i)$ for $i = t - l_t + 1, ..., t$ where $l_t = k_t + n_F - 1$.

If the data is non-Gaussian due to a considerable number of outliers, one strategy is to simply truncate the values z to a pre-defined maximum. Such a truncation may also be a consequence of the A to D conversion. However the result will be very non-Gaussian in a different way, with undesirable results. One way to handle this is to replace the squared residual term in formula (3.4) by a mixed L_1, L_2 functional. In this case residual components below a threshold T are squared as before, but above T this is replaced by the absolute value, scaled and offset in such a way that continuity of position and slope is maintained:

$$|\mathbf{x}|_{L_1,L_2} = \sum x_i^2 \text{ for } |x_i| \leq T \text{ else } 2T|x_i| - T^2 \tag{3.9}$$

The L_1 functional puts much less weight on large outliers. In fact, it is well known to statisticians that large enough outliers can actually have no effect on the solution. Fortunately, our Newton-GMRES solver takes this in stride, which again shows why this approach is so flexible. The reason that a Newton method can still work is that the discontinuity is only in the second derivative (at T), and only a few out of many sample residuals cross this threshold between function evaluations. A smooth approximation to the L_1, L_2 functional, such as $\sum 2T{x_i}^2/(|x_i| + 2T)$, avoids this problem but at additional cost. A strategy for even stronger filtering is to simply delete entirely residual components above T, or to avoid irregularity in the indexing, use the smooth approximation $\sum T^2{x_i}^2/(x_i^2 + T^2)$ to residual truncation.

4. Newton-GMRES

GMRES [Ke] is now frequently used in the solution of pde's and ode's, but its application to signal processing and control is much more recent. We have developed a very general Newton-GMRES algorithm [Bu] that uses a Levenburg-Marquardt trajectory search for damping. This is an optimal form of the Levenburg-Marquardt "trust region" method [Gi, pp. 113-115].

Conjugate gradient type solvers such as GMRES are ideally suited for "well conditioned" problems that required fast approximate solution. Signal processing and control problems that are derived from the minimization of a functional are often so well conditioned that they don't need any of the special "pre-conditioning" that is normally required for pde's and ode's. Pre-conditioning amounts to applying approximate solution techniques, such as relaxation, that have the effect of clustering the eigenvalues of the Jacobian away from zero. The

Jacobian is the matrix that results from linearizing the equation, (3.5) in our application, by taking partials with respect to the unknowns at their current values. We use no pre-conditioning other than "diagonal preconditioning": divide each component equation to be solved by the corresponding diagonal element of the Jacobian. Note that this solves the equation exactly if the Jacobian is itself diagonal.

What makes this relatively sophisticated approach practical is the used of batch processing. That is, the overhead associated with GMRES is small compared to the cost of the IIR filters when the batch includes tens, even hundreds, of time samples. Likewise the cost the of the Shur-Cohn stability test for IIR filters, also referred to as Jury's test, is small in this case.

In the Newton-GMRES method, a quadratic approximation to the functional is formed on a small dimensional "Krylov subspace", instead of the full space, at each Newton step. For our active noise control the dimension is typically 1 to 5, when the dimension of the full space may be in the hundreds. Each dimension, or iteration, requires one directional derivative computation, which is normally achieved by a finite difference using one function evaluation (the IIR filters).

Generically, we represent the equations to be solved, such as (3.5), by

$$F(\mathbf{x}) = 0 \tag{4.1}$$

where $\mathbf{x}$ is a vector of n unknowns and F represents n nonlinear functions. The algorithm begins by calling the user routine to evaluate F at the initial estimate $\mathbf{x}_0$. The corresponding initial sum of squares residual is computed as

$$r(\mathbf{x}_0) = \frac{1}{2}|F(\mathbf{x}_0)|^2 \tag{4.2}$$

GMRES proceeds to minimize the residual $r(\mathbf{x})$ over a Krylov subspace of dimension k. The Krylov subspace is defined formally by

$$\mathbf{e}, J\mathbf{e}, J^2\mathbf{e}, ..., J^{k-1}\mathbf{e} \tag{4.3}$$

where $\mathbf{e} = F(\mathbf{x})$ and the Jacobian J is defined by

$$J = \left(\frac{\partial}{\partial \mathbf{x}} F\right)(\mathbf{x}_0) \tag{4.4}$$

In practice, instead of computing the Jacobian matrix, computations such as $J\mathbf{e}$ are regarded as directional derivatives to be computed by the finite difference formula

$$J\mathbf{e} = F_{\mathbf{e}}(\mathbf{x}) = (F(\mathbf{x} + \varepsilon\mathbf{e}) - F(\mathbf{x}))/\varepsilon \tag{4.5}$$

The "Arnoldi process" obtains an orthonormal basis - the search directions, or Arnoldi basis - for the Krylov subspace by the modified Gram-Schmidt procedure:

$$
\begin{aligned}
\mathbf{w}_1 &= F(\mathbf{x}); \quad \mathbf{v}_1 = \mathbf{w}_1/|\mathbf{w}_1| \\
\mathbf{w}_2 &= F_{\mathbf{v}_1} - (F_{\mathbf{v}_1} \cdot \mathbf{v}_1)\,\mathbf{v}_1; \quad \mathbf{v}_2 = \mathbf{w}_2/|\mathbf{w}_2| \\
&\ldots \\
\mathbf{w}_k &= F_{\mathbf{v}_{k-1}}, \ldots, \ \mathbf{w}_k = \mathbf{w}_k - (\mathbf{w}_k \cdot \mathbf{v}_j)\,\mathbf{v}_j, \ldots; \quad \mathbf{v}_k = \mathbf{w}_k/|\mathbf{w}_k|
\end{aligned}
\tag{4.6}
$$

The minimization of r on the Krylov subspace $V = (\mathbf{v}_1, \ldots, \mathbf{v}_k)$ is accomplished by solving the normal equations for the standard Newton step $\mathbf{s}_1$:

$$
G\mathbf{s}_1 = -\mathbf{g} \tag{4.7}
$$

where $\mathbf{g}_j = r_{\mathbf{v}_j}(x) = F(x) \cdot F_{\mathbf{v}_j}(x)$ is the gradient and $G_{ij} = F_{\mathbf{v}_i} \cdot F_{\mathbf{v}_j}$. The solution uses a reformulation in terms of an upper Hessenburg matrix H:

$$
H_{ij} = F_{\mathbf{v}_j} \cdot \mathbf{v}_i; \quad G = H^t H \tag{4.8}
$$

and a numerically stable Givens rotation procedure to reduce H to upper triangular. As a by-product, the minimal value of r is known at each stage of this procedure, and if it falls below the user-specified stopping tolerance, or to the level of numerical noise, then the procedure is stopped, resulting in fewer than k search directions.

The Levenburg-Marquardt trajectory for Newton damping is achieved by replacing G in equation (4.7) by $G + \lambda(t)I$ to get the trajectory vector $\mathbf{s}_t$ by another QR factorization:

$$
\begin{aligned}
G_t &= H^t H + \lambda(t) I \\
&= R_0^t R_0 + \lambda I \\
&= (RE)^t (RE) \text{ for } RE = \begin{pmatrix} R_0 \\ \sqrt{\lambda} I \end{pmatrix} \\
&= (Q_\lambda R_\lambda)^t (Q_\lambda R_\lambda) = R_\lambda^t R_\lambda
\end{aligned}
\tag{4.9}
$$

Here $\lambda(t)$ is proportional to $(1-t)/t^2$ for $0 < t \leq 1$, so that for $t \sim 0^+$ we get $G_t \sim \lambda(t) I$ and s_t is a negative gradient, or steepest descent, direction. Then a line search is used to minimize either the original functional or $r(\mathbf{x}_t)$ over t where $\mathbf{x}_t = \mathbf{x}_0 + V\mathbf{s}_t$ with $\mathbf{s}_t \equiv t\,\mathbf{s}_1$ for any values of $t > 1$ that are to be included. Thus as long as descent is possible, it is guaranteed, and, in fact, the maximum possible trust region descent is achieved on a continuum between the steepest descent direction and the Newton direction.

After initially bracketing the minimum using increasing steps in t, the line search does quadratic interpolation on a segment to estimate the location of the minimum. This interpolation uses a function value at one end point and the function value and its derivative at the other endpoint (this finite difference derivative was already computed for the bracketing process). Quadratic interpolation was found to be more robust than a cubic, especially for a mixed L_1, L_2 functional.

The decision as to whether or not damping is needed is based on the prediction ratio: The ratio of the actual decrease in r at the Newton step s_1 to the predicted decrease in r, the latter being a by-product of the solution for s_1. If this ratio is close to 1, say > 0.75, then the functional is close to quadratic and no damping is needed.

A major problem with the damping procedure just outlined is that it is possible that the function may blow up, or not be defined, at the Newton step or other steps of the trajectory. In the active noise control application this can arise from unstable IIR filters. Therefore it is important to first check the coefficients x_{t_0} at the initial value t_0 of the trajectory search before proceeding. This is where the Shur-Cohn test comes in. If the test fails, then just decrease t_0 by a suitable step until it succeeds by at least a small margin. Then a trajectory search on $(0, t_0)$ must succeed. Typically, a line search takes 3 to 4 function evaluations with a goal of 10% accuracy, but there can be considerable variation.

For IIR filters, we start with $\mathbf{x_0} = 0$ and first converge the FIR coefficients for a couple of updates. Then we might start with an initial value of $t_0 = 0.75$, say, for estimation of the IIR coefficients. After an update, t_0 is increased by 0.1 from its final value on the last update, up to a maximum value of 1.0. This strategy for adjusting the initial Levenburg-Marquardt parameter applies to a wide variety of problems.

5. Conclusion

For many applications we no longer need restrict our control and signal processing algorithms to the very simplified or very clever linear methods and special purpose hardware of the past. This has been demonstrated by our application of a Newton-GMRES algorithm to multi-dimensional active noise control using IIR filters. Lab tests typically get 10dB reduction of broad band noise below 1000 Hz in a cluttered room with one speaker per feedback sensor and one source sensor per physical source. However, it is also hard to get this kind of reduction, even with many source sensors, if the physical source is very diffuse (distributed), or if travel times from source to quiet zone are

not less than mechanical delay times of the actuators. These tests used a standard DSP board with C compiler, using code converted from the original Fortran 90. Debugging the C code was more onerous than necessary, pointing to the need for better language support.

The key to this new paradigm is batch processing for coefficient updates, while real-time processes run in the background. Then the overhead associated with modern mathematical algorithms is small. In particular our Newton-GMRES code is very efficient, flexible, robust, and widely applicable to both linear and nonlinear problems. Coupled with tests for the validity of a possible set of coefficients, such as the Shur-Cohn test for IIR filter stabililty, this code has guaranteed reliability.

In its simplest usage, with only one search direction, GMRES is a gradient algorithm with optimal estimation of the scaling coefficient. As such it has the robustness properties of the LMS method [Ha]. With a few more search directions the coefficients usually converge faster, but it is actually possible to over solve for coefficients that might fit a limited segment of data very well but not a longer segment. Thus a full nonlinear least squares solution would not be desirable even if it were free. Five search directions was the maximum we found useful for active noise control.

References

[Bu] R.H. Burkhart, *GMRES Acceleration and Optimization Codes*, Boeing Computer Services Technical Report, ETA-TR-88R1, 1989.

[El] D.F. Elliott, *Handbook of Digital Signal Processing - Engineering Applications,* Academic Press, San Diego.

[Fl] R. Fletcher, *Practical Methods of Optimization: Unconstrained Optimization*, Wiley, 1980.

[Gi] P.E. Gill, W. Murray, and M. Wright, *Practical Optimization*, Academic Press, London, 1981.

[Ha] B. Hassibi, A.H. Sayed, T. Kailath, H^∞ *Optimality of the LMS Algorithm*, IEEE Transactions on Signal Processing, vol. 44, No. 2, February, 1996.

[Ke] C.T. Kelley, *Iterative Methods for Linear and Nonlinear Equations*, SIAM, Philadelphia, 1995.

[Ne] P.A. Nelson and S.J. Elliott, *Active Control of Sound*, Academic Press, San Diego, 1992.

[Wi] B. Widrow and S.D. Stearns, *Adaptive Signal Processing*, Prentice Hall, Englewood Cliffs, NJ, 1985.

Boeing Information and Support Systems, P.O. Box 3707, MS 7L-21, Seattle, WA 98124

E-mail address: burkhart@rt.cs.boeing.com

GRAY CONNECTED COMPONENTS FOR STEREO CORRESPONDENCE PROBLEM

Yang Wang and Prabir Bhattacharya

Department of Computer Science and Engineering
University of Nebraska-Lincoln, Lincoln, NE 68588–0115
Tel: (402) 472-5002, Fax: (402) 472-7767
e-mail: ywang@cse.unl.edu, prabir@cherokee.unl.edu

ABSTRACT

In this paper. we investigate using the boundary of a parameter-dependent connected component of an image to solve the stereo correspondence problem. Under some conditions, we find a one-to-one relationship between the boundary points of the components of the stereo images which could be identified through the standard epipolar geometry. Using the relationship, we may also recover the surface corresponding to a connected component hierarchically.

Key Words: stereo images, correspondence problem, feature-based matching, connected components, hierarchical matching.

1 INTRODUCTION

The depth information of a 3-D object is essential in many applications. For a pair of stereo images, if we could determine which object in one image corresponds to a given object in the other image, then principles of stereopsis enable us to compute the depth of this object.[1] There is an extensive literature about the correspondence problem.[2] They may be classified as: *point-based matching* or *feature-based matching*. Point-based matching[3–5] will confront a difficulty if an object surface appears with different foreshortenings. However, there are often some features in an image that can be identified and utilized in an appropriate manner for image matching. A variety of features, including zero-crossings,[6] gray-level "corner",[7] lines,[8–11] curve,[12] shape,[13] has been used for the correspondence problem. Among these, the line-based methods appear to be the most popular. However, the line-based methods provide only sparse depth information and interpolation should be employed. Also the lines themselves usually do not represent any meaningful objects in an image. In order to recover the surface of an object, one needs to group the lines in a certain way – this is, however, not easy in many applications and ambiguity is a serious problem.

In this paper, we describe a new feature-based method for the correspondence problem. The feature that we use is the boundary of an (ϵ, δ)-*components*[14–17] of an image. Under some constraints, there is a one-to-one relationship between the sets of the boundary points of an (ϵ, δ)-component on the conjugate epipolar lines. Using this relationship, we may find the corresponding boundaries of the (ϵ, δ)-components of the stereo images. Consequently, we may obtain, uniquely, the corresponding (ϵ, δ)-components represented by the corresponding

boundaries. Since only the boundary points have accurate correspondence, the surface corresponding to an (ϵ, δ)-component in the 3-D space are not determined uniquely. However, by decreasing the values of ϵ or δ gradually, we obtain a hierarchy of the (ϵ, δ)-components, and each (ϵ, δ)-component at a lower level is a sub-area of an (ϵ, δ)-component at a higher level. Thus, we also obtain a hierarchy of the boundaries of the (ϵ, δ)-components. Finding the correspondence of the boundaries at a lower level increases the known corresponding points on the surface. So, the surface may be determined more accurately.

2 STEREO MATCHING OF CONNECTED COMPONENTS

To make the paper self-contained we review briefly some definitions from.[14] Given two nonnegative integers ϵ and δ, two points p, q in a gray image Σ are said to be (ϵ, δ)-*connected* if there exists a path $\rho : p = p_0, p_1, \ldots, p_n = q$, $n \geq 0$, such that the maximum variation of the gray values of all the points along the path is less than or equal to ϵ, and the variation of the gray values between the successive points p_i and p_{i+1} $(0 \leq i \leq n-1)$ is less than or equal to δ. A subset of Σ is called an (ϵ, δ)-*connected set* if each pair of points of the subset is (ϵ, δ)-connected. Given a point $p \in \Sigma$, any maximal (ϵ, δ)-connected set containing p is called a *related* (ϵ, δ)-*connected component* ((ϵ, δ)-RCC) of p. In[14] we have given an algorithm to partition an image into a group of (ϵ, δ)-RCCs. Each of the (ϵ, δ)-RCC is called an (ϵ, δ)-*component* of the image.

2.1 Matching (ϵ, δ)-component boundary

Our objective is to find the corresponding boundary in one image for a given boundary of an (ϵ, δ)-component in the other image. In general, it is not possible to find a one-to-one correspondence between the points in two stereo images due to foreshortening. If the stereo images satisfy following constraints: (1) any boundary point that is visible from one camera should be visible from the other camera; (2) if there are multiple objects in the scene, the two cameras should be placed reasonably close so that the images of the objects have the same relative positions in both stereo images; (3) the effect of foreshortening will not cause non-adjacent boundary points in one image to be adjacent in the other image; then we show that there is a one-to-one correspondence between the sets of the boundary points of the corresponding (ϵ, δ)-components.

We shall use the superscripts or subscripts "l", "r" to represent parameters referring to the left or right camera respectively. Without loss of generality, we may assume that the boundary of a given (ϵ, δ)-component is on the image I_l. For each point p^l on the given boundary, we can find its conjugate epipolar lines E_l and E_r. According to our first assumption, the point p^r should also be a boundary points along the line E_r. Due to foreshortening, there may be more than one point on E_r that has p^l as its conjugate point. These points themselves should be connected to each other on the line E_r. Similarly, a boundary point on E_r may have more than one connected boundary point on the line E_l as its conjugate point. We shall refer to each group of connected boundary points on an epipolar line as a *connected set.*

Since either the boundary points or the connected sets are distributed only on an epipolar line, we are able to label them in the same order as that on its conjugate epipolar line from the left end to the right end relative to the corresponding optical center.

Propositions:

1. *The relative positions of any two boundary points remain the same on the conjugate epipolar lines.*

2. *For a connected set S on an epipolar line, the set S' of the conjugate points of the points in S is a connected set on the conjugate epipolar line. (We shall call the sets S, S' the* corresponding connected sets.*)*

3. *The number of connected sets on two conjugate epipolar lines are equal.*

4. *The relative positions of any two connected sets remain the same on the conjugate epipolar lines.*

From Propositions 2–3, it follows that there is a one-to-one correspondence between the connected sets on the conjugate epipolar lines. Combining Propositions 1–4, we obtain the following result which is the basis of our algorithm for finding the corresponding boundaries of the (ϵ, δ)-components.

THEOREM 1.

Let E_l, E_r be the two conjugate epipolar lines, $l_1, l_2, \ldots, l_m$ be the connected sets on E_l, $r_1, r_2, \ldots, r_m$ be the connected sets on E_r. If a boundary point p^l belongs to l_i, $1 \leq i \leq m$, then its conjugate point p^r belongs to r_i.

Using Theorem 1, for a given boundary point, we can determine uniquely the connected set that contains the conjugate point on the other image. Since the points on the boundary of an (ϵ, δ)-component are adjacent in order, their corresponding connected sets are also adjacent in order. Therefore, the points of these connected sets form the boundary of the corresponding (ϵ, δ)-component.

Using the following algorithm whose time complexity is $m \times n \times \max(m, n)$ for an $m \times n$ image, we find a corresponding boundary pair of the (ϵ, δ)-components and then, we obtain the corresponding (ϵ, δ)-components limited by the boundary pair.

ALGORITHM 1.

Let $B^l = \{p_1^l, p_2^l, \ldots, p_k^l\}$ be a given boundary of an (ϵ, δ)-component on the image I_l where $p_1^l, p_2^l, \ldots, p_k^l$ are the points on the boundary, and B^r be the corresponding (ϵ, δ)-component's boundary in the image I_r.

0. $B^r = \{\}$;
1. for $i = 1$ to k do
2. { determine the epipolar lines E_l, E_r of p_i^l;
3. find the connected set l_j such that $p_i^l \in l_j$ on E_l;
4. find the connected set r_j on E_r;
5. $B^r = B^r \cup r_j$;
   }

2.2 Matching (ϵ, δ)-component hierarchy

Using Algorithm 1, we match each (ϵ, δ)-component in a pair of stereo images. However, the points within an (ϵ, δ)-component are not always matched in a one-to-one manner – only the points on the boundary of an (ϵ, δ)-component may have accurate correspondence. So, the 3-D surface represented by a pair of corresponding (ϵ, δ)-components may not be reconstructed uniquely.

From the definition of (ϵ, δ)-components, we know that when $\epsilon > 0$, $\delta > 0$, the gray values of the points within an (ϵ, δ)-component are usually different. If we choose an $\epsilon' < \epsilon$ or $\delta' < \delta$, then an (ϵ, δ)-component C can be decomposed further into a set of (ϵ', δ')-components of C. So, if C^l and C^r are identified as the corresponding (ϵ, δ)-components in a pair of stereo images, then after finding their (ϵ', δ')-components, we may use the same method to identify the corresponding (ϵ', δ')-components of C^l and C^r. The points on the boundaries of the (ϵ', δ')-components are within C^l or C^r and have accurate correspondence. So, we could find more corresponding points of the surface determined by C^l and C^r. This procedure may be repeated for each pair of corresponding (ϵ', δ')-components until $\epsilon = 0$ or $\delta = 0$, in which case, the points of a connected component have the same gray

value. By decreasing the value of ϵ or δ gradually, the sizes of the (ϵ, δ)-components become smaller and smaller, and more and more corresponding points of the surface are identified. Therefore, the surface of a 3-D object may be determined uniquely up to the patches that have a single gray value. Since we may treat an original image as one (ϵ, δ)-component with the values of ϵ and δ to be the maximal gray value of the image,[14] and each (ϵ, δ)-component obtained within the repeated process depends only on an (ϵ, δ)-component determined at the previous iteration, all the (ϵ, δ)-components computed in the matching process form a 3-D-tree.

The following are certain advantages in matching the (ϵ, δ)-components hierarchically: (1) Since an (ϵ, δ)-component at a lower level is a subset of an (ϵ, δ)-component dealt with at a higher level, matching at a higher level of the hierarchy constrains the matching at lower levels. (2) In some applications, it is not necessary to find all the corresponding points of the surface of an object – we may need to match the surface patches only up to a certain detail that may be represented by the (ϵ, δ)-components on the image of the surface. Decreasing the values of ϵ or δ, the patches gradually get smaller, and more points on the surface get matched. So, with the hierarchy, we may easily control how accurately the surface be determined or decide when the matching result is satisfactory.

3 IMPLEMENTATION

Our experiments were carried out on a variety of stereo images using a SUN 4 workstation running the standard *Khoros* image processing environment. All experimental images shown in this paper are size of 300×300 pixels. We use white curves to indicate the boundaries of (ϵ, δ)-components and black components indicate a matched pair of (ϵ, δ)-components. Figure 1 shows an example of finding the corresponding (ϵ, δ)-components. Figure 1-(a) are the original stereo images. Figure 1-(b) shows the (ϵ, δ)-components of the images in Figure 1-(a) when $\epsilon = 12$, $\delta = 5$. We see that there are two (ϵ, δ)-components on each image. Figure 1-(c) shows the matched (ϵ, δ)-components (the (ϵ, δ)-component at the right of each image). Figure 2 shows an example of finding the corresponding (ϵ, δ)-components hierarchically. Figure 2-(a) are the original stereo images. Figure 2-(b) shows the (ϵ, δ)-components of Figure 2-(a) when $\epsilon = 165$, $\delta = 20$. There are two (ϵ, δ)-components on each image – one is composed of the darkest part of the image, and the other is composed of the remaining part of the image. Here, the darker (ϵ, δ)-component in one image is matched with the darker (ϵ, δ)-component component in the other image, and the lighter (ϵ, δ)-component is matched with the other lighter component. Now, we consider the points within the lighter component pair and ignore the points within the darker components. By decreasing the values of ϵ and δ to 100 and 10 respectively, we obtain the (ϵ, δ)-components of the lighter components as shown in Figure 2-(c), in which each lighter component is decomposed further into three (ϵ, δ)-components. Using the same matching method, we may find their corresponding component pairs. Figure 2-(d) shows one of the matched component pair (black areas). Repeating this procedure, we then may decompose the known component pair in the previous stage further as shown in Figure 2-(e), (g), and obtain the corresponding component pairs as shown in Figure 2-(f), (h). From this example, we see that by decreasing the values of ϵ and δ, the matched patches become smaller and smaller which in turn increases the accuracy of recovering the surface.

4 CONCLUSIONS

We have proposed a method to find the corresponding (ϵ, δ)-components of stereo images by matching their boundaries. Through the hierarchy of the (ϵ, δ)-components of an image, we obtain a stereo matching hierarchy in which we may conveniently adjust the accuracy of a reconstructed surface according to the needs of an application. The proposed method combines the epipolar geometry with feature matching. So, it has a lower computational complexity than a pure feature matching method, and has a higher confidence in matching than a pure epipolar geometry method.

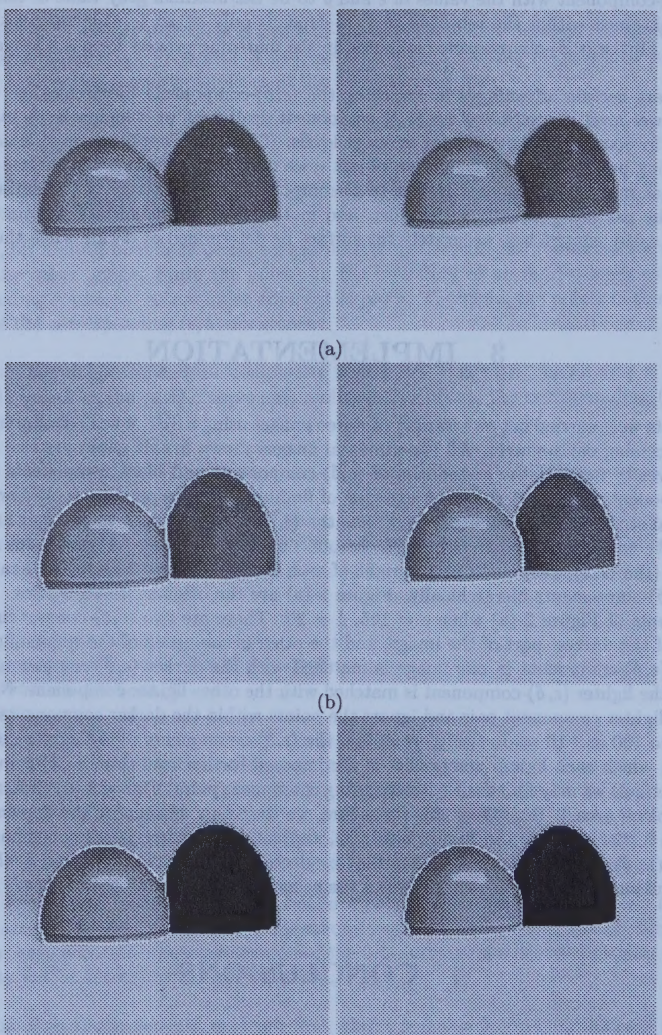
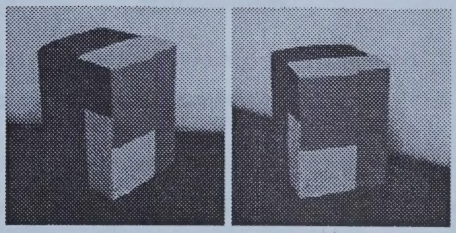
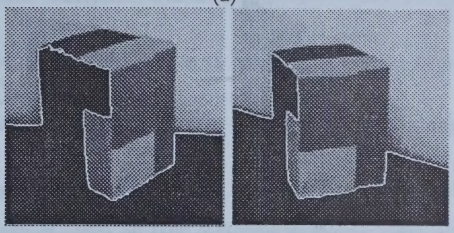


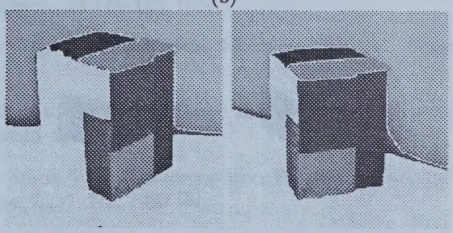
Figure 1: (a) A stereo image pair. (b) The (ϵ, δ)-components of the stereo images when $\epsilon = 12$, $\delta = 5$. (c) A corresponding (ϵ, δ)-component pair (black areas).



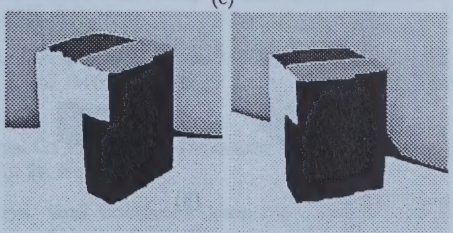
(a)



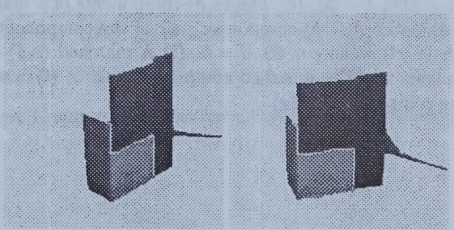
(b)



(c)



(d)



(e)

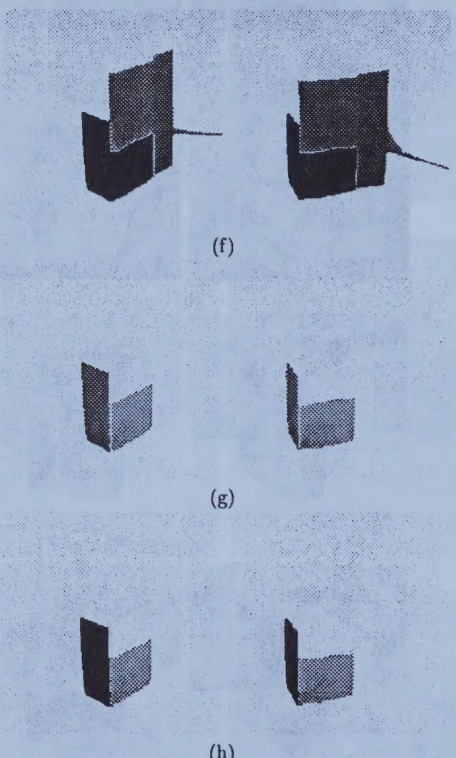


Figure 2: An example of matching component hierarchy. (a) The original stereo images. (b) The (ϵ, δ)-components of the images in (a) when $\epsilon = 165$, $\delta = 20$. (c) The (ϵ, δ)-components of a matched component pair shown in (b) when $\epsilon = 100$, $\delta = 10$. (d) A matched (ϵ, δ)-component pair of the components in (c). (e) The (ϵ, δ)-components of the matched components in (d) when $\epsilon = 80$, $\delta = 5$. (f) A matched (ϵ, δ)-component pair of the components in (e). (g) The (ϵ, δ)-components of the matched component pair in (f) when $\epsilon = 50$, $\delta = 3$. (h) A matched (ϵ, δ)-component pair of the components in (g).

References

[1] O. Faugeras. *Three-Dimentional Computer Vision.* MIT Press, Cambridge, MA, 1993.

[2] U. R. Dhond and J. K. Aggarwal. Structure from Stereo – A Review. *IEEE Trans. Syst., Man, Cybern.*, 19:1489–1510, 1989.

[3] S. T. Barnard and M. A. Fischler. Computational Stereo. *Computing Surveys*, 14:553–572, 1982.

[4] D. Marr and T. Poggio. Cooperative Computation of Stereo Disparity. *Science*, 194:283–287, 1976.

[5] B. K. Opitz. Advanced Stereo Correlation Research. *Photogrammetric Engineering and Remote Sensing*, 49:533–534, 1983.

[6] W. E. L. Grimson. Computational Experiments with a Feature Based Stereo Algorithm. *IEEE Trans. Patt. Anal. Mach. Intell.*, 7:17–34, 1985.

[7] B. K. P. Horn. Non-Correlation Methods for Stereo Matching. *Photogrammetric Engineering and Remote Sensing*, 49:535–536, 1983.

[8] N. Ayache. *Artificial Vision for Mobile Robots.* MIT Press, Cambridge, MA, 1991.

[9] Z. N. Li. Stereo Correspondence Based on Line Matching in Hough Space Using Dynamic Programming. *IEEE Trans. Syst., Man, Cybern.*, 24:144–152, 1994.

[10] G. Medioni and R. Nevatia. Segment Based Stereo Matching. *Comput. Vision, Graphics, and Image Processing*, 31:2–18, 1985.

[11] V. Venkateswar and R. Chellappa. Hierarchical Stereo and Motion Correspondence Using Feature Groupings. *Int. J. Computer Vision*, 15:245–269, 1995.

[12] N. M. Nasrabadi and Y. Liu. Stereo Vision Correspondence Using a Multichannel Graph Matching Technique. *Image and Vision Computing*, 7:237–245, 1989.

[13] S. Sclaroff and A. P. Pentland. Modal Matching for Corresponence and Recognition. *IEEE Trans. Patt. Anal. Mach. Intell.*, 17:545–561, 1995.

[14] Y. Wang and P. Bhattacharya. A Theory of Parameter–Dependent Connected Components of Gray Images and Segmentation. In *Proc. IEEE Internat. Conf. on Image Processing*, volume 3, pages 69–72, Washington DC., Oct. 1995.

[15] Y. Wang and P. Bhattacharya. On Parameter – Dependent Connected Components of Gray Images. *Pattern Recognition*, To appear in Vol 29, No. 8, 1996.

[16] Y. Wang and P. Bhattacharya. Gray Connected Components and Image Segmentation. To appear in the *Proc. SPIE Conf. Applications of Digital Image Processing XIX*, Denver, CO., Aug. 1996.

[17] Y. Wang and P. Bhattacharya. Image Analysis and Segmentation Using Gray Connected Components. To appear in the *Proc. IEEE Internat. Conf. on Systems, Man and Cybernetics*, Beijing, China, Oct. 1996.

[18] B. K. P. Horn. *Robot Vision.* The MIT Press, Cambridge, MA, 1986.

[19] Y. Wang and P. Bhattacharya. Connected Components, Segmentation and Image Understanding. Submitted for publication, 1996.

Delta-Operator Linear Prediction Models for Two-Dimensional Data*

Lih-Huah Yiin and H. Vincent Poor
Electrical Engineering Department
Princeton University
Princeton, NJ 08544
USA

Abstract

Delta-operator-based models and algorithms have been shown to be numerically superior to conventional shift-operator models for one-dimensional signals sampled at high sampling rates. In this paper, we will extend this technique to the problem of modeling densely-sampled two-dimensional data with spatial delta operators. A linear prediction model with quarter-plane support is formulated by the delta operator. Algorithms for estimating model parameters from stationary data and live data are presented. These algorithms for delta-operator models can achieve the same computational efficiency as the algorithms for shift-operator models. A computer simulation demonstrates the numerical advantages of delta-operator models.

1 Introduction

The use of difference equations in characterizing two-dimensional random fields has various applications in image processing, e.g. image coding [1] and texture analysis [2]. The estimation of model parameters from observed data is an important part in such applications. However, as pointed out in [3], parameter estimation for shift-operator models is generally ill-conditioned when the data are highly correlated. In recent years, a delta-operator approach has been shown to be effective in alleviating the ill-conditioning encountered in one-dimensional signal processing [3] [4]. Transforming two-dimensional difference equation models into delta-operator form may improve the numerical accuracy in this problem as well.

Delta-operator models are also motivated from partial-differential-equation (PDE) models for continuous random fields. In [5] and [6], second-order PDEs – parabolic, elliptic, and hyperbolic – are used to model two-dimensional continuous fields. In [7], these PDEs are discretized by finite difference approximation method, and the resulting difference equations are used to model images in applications such as image restoration. Some particular forms of higher-order PDE modeling two-dimensional fields are also

*This work was supported by the U.S. Office of Naval Research under Grant N00014-94-1-0115.

possible [8]. To investigate the connection between the continuous and discrete models with the delta operator formulation will also be of interest.

To study these issues, we first focus on the linear prediction models with quarter-plane support, because it is a more straightforward extention of the one-dimensional linear prediction model. We will derive efficient algorithms for computing the model parameters for stationary data. We will also derive a fast adaptive algorithm for updating the model parameters from newly-arriving data. We further consider the behavior of these models in the limit of infinitely dense sampling, and we present simulation results to verify the predicted performance improvement afforded by the delta-operator formulation.

2 Quarter-Plane Linear Prediction Models

Let $\{x(k,l); k,l \in \mathcal{Z}\}$ be a real homogeneous random field with zero mean. The two-dimensional linear prediction model of $\{x(k,l)\}$ is given by the representation

$$e(k,l) = x(k,l) + \sum\sum_{(i,j)\in\mathcal{N}} a_{N,M}[i,j]x(k-i,l-j) \tag{2.1}$$

where $\mathcal{N} = \{(i,j) : 0 \le i \le N, 0 \le j \le M, (i,j) \ne (0,0)\}$ is the quarter-plane prediction window as shown in Figure 2.1. The desired coefficients $a_{N,M}[i,j]$'s can be determined by minimizing $E[e^2(k,l)]$, that is, from the normal equations

$$\mathbf{R}_{N,M}\underline{a}_{N,M} = \sigma^2[1,0,\cdots,0]^T \tag{2.2}$$

where

$$\begin{aligned}
\mathbf{R}_{N,M} &= E[\vec{X}_{N,M}(k,l)\vec{X}^T_{N,M}(k,l)], \\
\vec{X}_{N,M} &= [x(k,l), x(k,l-1), \cdots, x(k,l-M), \\
&\quad x(k-1,l), x(k-1,l-1), \cdots, x(k-1,l-M), \\
&\quad \vdots \\
&\quad x(k-N,l), x(k-N,l-1), \cdots, x(k-N,l-M)]^T, \\
\underline{a}_{N,M} &= [a_{N,M}[0,0], a_{N,M}[0,1], \cdots, a_{N,M}[0,M], \cdots, a_{N,M}[N,0], \cdots, a_{N,M}[N,M]]^T,
\end{aligned}$$

and

$$\sigma^2 = \min_{\underline{a}_{N,M}} E[e^2(k,l)].$$

Suppose $\{x(k,l)\}$ is sampled from a continuous field $\{X(t_1,t_2); t_1,t_2 \in \mathcal{R}\}$. Define the forward delta operator

$$\delta = \frac{q_1 - 1}{\Delta}; \quad q_1\text{: shift operator, i.e. } q_1 x(k,l) = x(k+1,l)$$

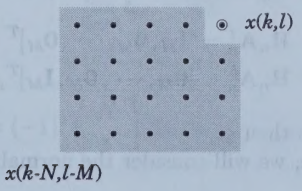


Figure 2.1: The shaded region is the quarter-plane prediction window $\mathcal{N}$

where Δ is the sampling interval. The delta-operator model represents $\delta^N x(k-N,l)$ as a linear function of the differenced data vector

$$\begin{aligned}\vec{W}_{N,M}(k,l) = [&\delta^N x(k-N,l-1),\cdots,\delta^N x(k-N,l-M),\\ &\delta^{N-1}x(k-N,l),\cdots,\delta^N x(k-N,l-M),\\ &\vdots\\ &x(k-N,l),\cdots,x(k-N,l-M)]^T\end{aligned}$$

with the coefficients $\tilde{a}_{N,M}[i,j]$'s. The model parameters $\underline{\tilde{a}}_{N,M}$ can be obtained by solving the equation

$$\mathbf{Q}_{N,M}\underline{\tilde{a}}_{N,M} = [\tilde{\sigma}^2, 0, \cdots, 0]^T \tag{2.3}$$

where

$$\mathbf{Q}_{N,M} = E[\vec{Z}_{N,M}(k,l)\vec{Z}_{N,M}(k,0)^T] = \mathbf{T}_{N,M}^T \mathbf{R}_{N,M}\mathbf{T}_{N,M},$$

and $\underline{\tilde{a}}_{N,M}$ is related to $\underline{a}_{N,M}$ by

$$\underline{\tilde{a}}_{N,M} = \Delta^{-N}\mathbf{T}_{N,M}^{-1}\underline{a}_{N,M}$$

where the transform matrix $\mathbf{T}_{N,M}$ is an $(N+1)\times(N+1)$ lower-triangular block matrix, with block size $(M+1)\times(M+1)$. Its (k,l)-th block is given by

$$\frac{(-1)^{k-l}}{\Delta^{N-l}}\begin{pmatrix} N-l \\ k-l \end{pmatrix}\mathbf{I}_M; \quad 0 \le k,l \le N \tag{2.4}$$

where $\mathbf{I}_M$ denotes the identity matrix.

3 Parameter Estimation

In this section, we present an order-recursive algorithm for computing the model parameters of the delta-operator model described in the preceding section, provided that the covariance matrix of the data field is available.

An $\mathcal{O}(N^2M^3)$ algorithm solving (2.2) has been derived in [9]. This algorithm is recursive in N while M is fixed. For simplicity, we will drop M from the subscript in all the notations. For convenience, we will consider the normalized normal equations,

$$\mathbf{R}_n\underline{\alpha}_n = [1, 0, \cdots, 0]^T \tag{3.1}$$

where $\underline{\alpha}_n = \underline{a}_n/\sigma^2$. We need some auxiliary vectors in order to complete the recursive algorithm, that is, the vectors $\mathbf{A}_n^f$ and $\mathbf{A}_n^b$ which satisfy the following two sets of equations

$$\begin{aligned}\mathbf{R}_n\mathbf{A}_n^f &= [\mathbf{I}_M, \mathbf{0}_M, \cdots, \mathbf{0}_M]^T\\ \mathbf{R}_n\mathbf{A}_n^b &= [\mathbf{0}_M, \cdots, \mathbf{0}_M, \mathbf{I}_M]^T,\end{aligned}$$

and the first column of $\mathbf{A}_n^f$ is then $\underline{\alpha}_n$.

Similarly, for convenience, we will consider the normalized version of (2.3), namely,

$$\mathbf{Q}_n\underline{\tilde{\alpha}}_n = [1, 0, \cdots, 0]^T \tag{3.2}$$

where $\underline{\tilde{\alpha}}_n = \underline{\tilde{a}}_n/\tilde{\sigma}^2$ and

$$\underline{\tilde{\alpha}}_n = \Delta^n \mathbf{T}_n^{-1} \underline{\alpha}_n. \tag{3.3}$$

Let

$$\begin{aligned} \tilde{\mathbf{A}}_n^f &= \Delta^n \mathbf{T}_n^{-1} \mathbf{A}_n^f \\ \tilde{\mathbf{A}}_n^b &= \Delta^n \mathbf{T}_n^{-1} \mathbf{A}_n^b, \end{aligned}$$

and $\underline{\tilde{\alpha}}_n$ is the first column of $\tilde{\mathbf{A}}_n^f$. Then $\underline{\tilde{\alpha}}_N$ can be computed by the following recursive formula

$$[\tilde{\mathbf{A}}_n^f \quad \tilde{\mathbf{A}}_n^b]\Gamma_n = \Delta \left[\left(\Delta \begin{bmatrix} \tilde{\mathbf{A}}_{n-1}^f \\ \mathbf{0}_M \end{bmatrix} + \begin{bmatrix} \mathbf{0}_M \\ \tilde{\mathbf{A}}_{n-1}^f \end{bmatrix} \right), \begin{bmatrix} \mathbf{0}_M \\ \tilde{\mathbf{A}}_{n-1}^b \end{bmatrix} \right] \tag{3.4}$$

$$\Gamma_n = \begin{bmatrix} \mathbf{I}_M & \Gamma_n^b \\ \Gamma_n^f & \mathbf{I}_M \end{bmatrix}$$

$$\Gamma_n^f = \Delta^{-n+2}[\mathbf{0}_M, \cdots, \mathbf{0}_M, \mathbf{I}_M]\mathbf{Q}_n \begin{bmatrix} \tilde{\mathbf{A}}_{n-1}^f \\ \mathbf{0}_M \end{bmatrix} \tag{3.5}$$

$$\Gamma_n^b = \mathbf{J}_M (\Gamma_n^f)^T \mathbf{J}_M$$

for $n = 1, \cdots, N$. The order of computational complexity remains $N^2 M^3$.

Limiting Behavior

Now let us consider the limiting behavior of the coefficients in the two above models as the sampling intervals converge to zero. We make the following assumptions:

1. The partial derivatives of the continuous field $\{X(t_1, t_2)\}$ exist up to order $M + N$.
2. The covariance matrix $\mathbf{Q}_N^{(0)}$ of the vector $\vec{Z}_{N,M}^{(0)}$ is nonsingular, where $\vec{Z}_{N,M}^{(0)} \equiv \lim_{\Delta \to 0} \vec{Z}_{N,M}$ is composed of partial derivatives of $\{X(t_1, t_2)\}$.

Within the first assumption, the random fields under consideration satisfy smoothness condition. The second assumption is to ensure that the model is not over-determined.

Proposition 1 *Let $\underline{\tilde{\alpha}}_n^{(0)}$ be the solution to the equation*

$$\mathbf{Q}_N^{(0)} \underline{\tilde{\alpha}}_N^{(0)} = [1, 0, \cdots, 0]^T \tag{3.6}$$

Then

$$\lim_{\Delta \to 0} \underline{\tilde{\alpha}}_N = \underline{\tilde{\alpha}}_N^{(0)} \tag{3.7}$$

and

$$\lim_{\Delta \to 0} \Delta^{2N} \alpha_N[i, j] = (-1)^i \binom{N}{i} \tilde{\alpha}_N[0, j], \quad 0 \le i \le N, \ 0 \le j \le M. \tag{3.8}$$

Proof: From (3.2) and (3.6),

$$\begin{aligned}\|\underline{\tilde{\alpha}}_N - \underline{\tilde{\alpha}}_N^{(0)}\| &= \| \left(\mathbf{Q}_N^{-1} - (\mathbf{Q}_N^{(0)})^{-1}\right)[1, 0, \cdots, 0]^T \| \\ &\leq \|\mathbf{Q}_N^{-1}\| \|\mathbf{Q}_N^{(0)} - \mathbf{Q}_N\| \|(\mathbf{Q}_N^{(0)})^{-1}\|.\end{aligned}$$

Then (3.7) follows because

$$\lim_{\Delta \to 0} \mathbf{Q}_N = \mathbf{Q}_N^{(0)}.$$

(3.8) can be shown by noting the relation of $\underline{\alpha}_n$ and $\underline{\tilde{\alpha}}_n$ in (3.3). □

Therefore, the delta model parameters $\underline{\tilde{a}}_N$ or $\underline{\tilde{a}}_N$ converges to a limit dependent on the underlying continuous field. The shift model parameters

$$\underline{a}_N = \underline{\alpha}_N / \alpha_N[0, 0]$$

depend on the underlying field by those terms with higher power in Δ, and thus the dependency vanishes as $\Delta \to 0$.

4 Recursive Computation of Model Parameters

Delta operator based multichannel recursive least-squares (RLS) lattice algorithms were developed in [10] and [11]. In this section, we present a fixed-order time-recursive transversal filter type algorithm for efficiently computing 2-D least-squares linear prediction model coefficients over a two-dimensional finite deterministic field $\mathcal{S} = \{x(i, j); 0 \leq i \leq K, 0 \leq j \leq L\}$, where data arrive sequentially according to the scan pattern which goes from left to right and then from bottom to top. In deriving the delta-operator algorithm, we follow the same ordering scheme and methodology as those in [12].

The delta-operator algorithm is summarized in Table 4.1. Define $P = M + 1$ and $Q = (M + 1)(N + 1) - 1$ (number of 2-D predictor coefficients). This delta-operator algorithm requires PQ more operations than the algorithm in [12].

Simulation

As the first example, a two-dimensional homogeneous random field with Gaussian spectrum given by

$$S_X(\omega_1, \omega_2) = \frac{1}{2\pi} \exp\left\{-\frac{\omega_1^2 + \omega_2^2}{2}\right\}$$

is chosen as our underlying continuous field. We compute the model parameters using different degrees of precision, and then compare the relative error

$$\text{Relative error} \equiv \frac{|e_{fp}^2(n) - e_\infty^2(n)|}{e_\infty^2(n)}$$

at each iteration step, where $e_{fp}(n)$ is the instantaneous modeling error by finite precision computation and $e_\infty(n)$ is by full precision computation. The esemble-average relative error of 20 and 30 mantissa bits with $\Delta = 0.1$ are shown in Figure 4.1. The time-average relative errors versus different numbers of mantissa bits are plotted in Figure 4.2 for $\Delta = 0.1$ and $\Delta = 0.3$. Note that the delta-operator model is uniformly better than the corresponding shift-operator model in this situation, and the improvement is more significant for smaller sampling interval.

$$\underline{\tilde{e}}_f(n|n-1) = \left(\tilde{\mathbf{A}}^f(n-1)\right)^T \vec{W}_{N+1}(n) \tag{4.1}$$

$$\underline{\tilde{e}}_f(n) = \underline{\tilde{e}}_f(n|n-1)\gamma_N(n-1) \tag{4.2}$$

$$\tilde{\Sigma}_f^{-1}(n) = \tilde{\Sigma}_f^{-1}(n-1) - \frac{\tilde{\Sigma}_f^{-1}(n-1)\underline{\tilde{e}}_f(n)\underline{\tilde{e}}_f^T(n)\tilde{\Sigma}_f^{-1}(n-1)}{\gamma_N(n-1) + \underline{\tilde{e}}_f^T(n)\tilde{\Sigma}_f^{-1}(n-1)\underline{\tilde{e}}_f(n)} \tag{4.3}$$

$$\tilde{\mathbf{A}}^f(n) = \tilde{\mathbf{A}}^f(n-1) + \begin{bmatrix} \underline{0}_M \\ \underline{\tilde{g}}_N(n-1) \end{bmatrix} \underline{\tilde{e}}_f^T(n|n-1) \tag{4.4}$$

$$\underline{\tilde{g}}_{N+1}(n) = \begin{bmatrix} \underline{0}_M \\ \underline{\tilde{g}}_N(n-1) \end{bmatrix} + \tilde{\mathbf{A}}^f(n)\tilde{\Sigma}_f^{-1}(n)\underline{\tilde{e}}_f(n) \tag{4.5}$$

$$\underline{\tilde{m}}(n) = [\mathbf{0}, \cdots, I_M]\hat{\mathbf{T}}_{N+1}\underline{\tilde{g}}_{N+1}(n) \tag{4.6}$$

$$\underline{\tilde{e}}_b(n|n-1) = \left(\tilde{\mathbf{A}}^b(n-1)\right)^T \vec{W}_{N+1}(n) \tag{4.7}$$

$$\gamma_{N+1}(n) = \gamma_N(n-1) - \underline{\tilde{e}}_f^T(n)\tilde{\Sigma}_f^{-1}(n)\underline{\tilde{e}}_f(n) \tag{4.8}$$

$$\gamma_N(n) = \gamma_{N+1}(n-1)\left(1 - \underline{\tilde{e}}_b^T(n|n-1)\underline{\tilde{m}}(n)\right)^{-1} \tag{4.9}$$

$$\Delta\begin{bmatrix} \underline{\tilde{g}}_N(n) \\ \underline{0}_M \end{bmatrix} + \begin{bmatrix} \underline{0}_M \\ \underline{\tilde{g}}_N(n) \end{bmatrix} = \left(\underline{\tilde{g}}_{N+1}(n) + \tilde{\mathbf{A}}^b(n-1)\underline{\tilde{m}}(n)\right) \times \left(1 - \underline{\tilde{e}}_b^T(n|n-1)\underline{\tilde{m}}(n)\right)^{-1} \tag{4.10}$$

$$\tilde{\mathbf{A}}^b(n) = \tilde{\mathbf{A}}^b(n-1) + \left(\Delta\begin{bmatrix} \underline{\tilde{g}}_N(n) \\ \underline{0}_M \end{bmatrix} + \begin{bmatrix} \underline{0}_M \\ \underline{\tilde{g}}_N(n) \end{bmatrix}\right)\underline{\tilde{e}}_b^T(n|n-1) \tag{4.11}$$

$$\tilde{e}(n|n-1) = \vec{Z}_N(n)\underline{\tilde{a}}(n-1) \tag{4.12}$$

$$\underline{\tilde{a}}(n) = \underline{\tilde{a}}(n-1) + \begin{bmatrix} 0 \\ \underline{\tilde{g}}_N(n) \end{bmatrix}\tilde{e}(n|n-1) \tag{4.13}$$

Table 4.1: The delta-operator recursive-least-squares algorithm.

The second example is to model blocks from the actual image "Lena." The block size is 32x32. The blocks are numbered from left to right, and then top to bottom. Table 4.2 shows the average improvement made by delta-operator algorithm over different numbers of mantissa bits ranging from 20 to 30 bits. Though there are negative values, delta algorithm works better for the majority of the test images.

5 Conclusion

We have applied delta operators to two-dimensional linear prediction models. The delta-model parameters can be computed by algorithms with the same efficiency as the algorithms for shift-operator models. And the delta models have meaningful limit as the sampling density goes to infinity.

In this paper, we consider only causal models with a rectangular prediction window. For two-dimensional fields, there are other possibilities for further research, including noncausal models, or models with other shapes of support, e.g. diamond-shaped.

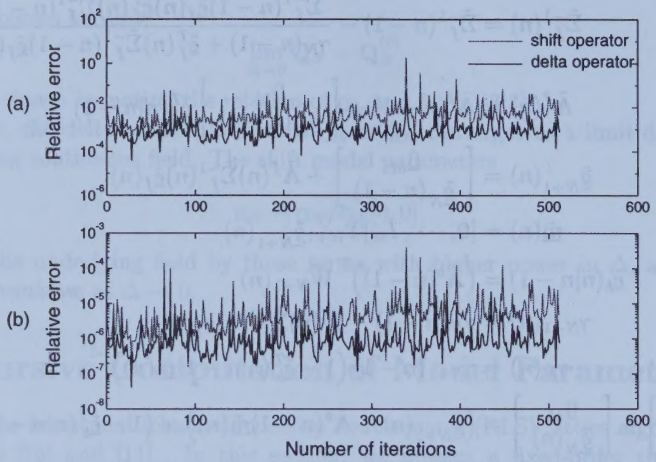
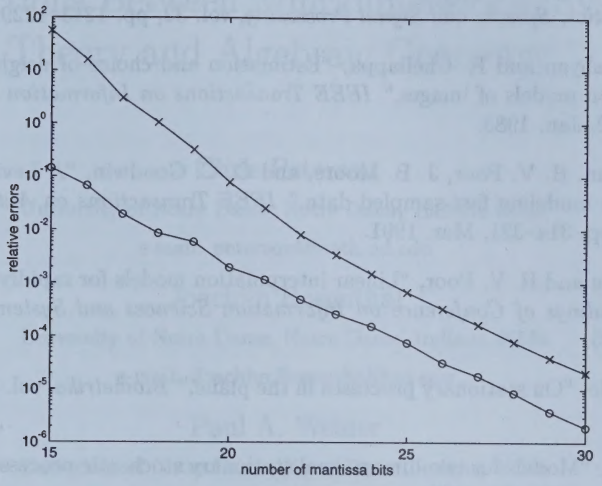


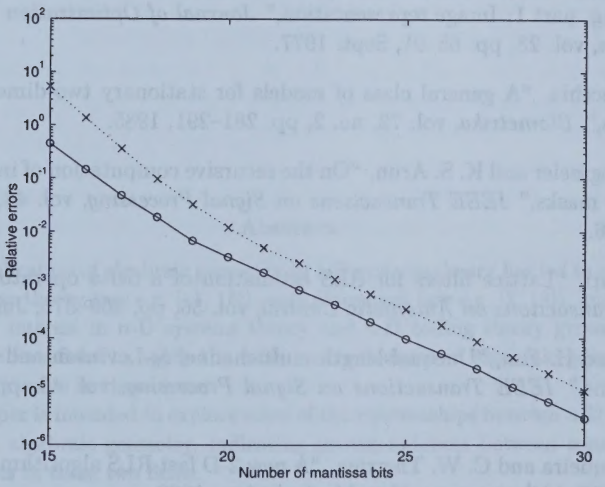
Figure 4.1: Ensemble-average relative error at each iteration. (a) 20 mantissa bits, (b) 30 mantissa bits, are used in the computation.

image	(dB)	image	(dB)
1	-	9	-0.38
2	4.30	10	9.87
3	6.46	11	12.69
4	-0.90	12	5.65
5	2.26	13	3.19
6	8.00	14	7.39
7	6.37	15	6.30
8	7.24	16	-5.38

Table 4.2: The average improvement of delta-operator algorithm on blocks from "Lena." The blocks are numbered from left to right, and then top to bottom.



(a)



(b)

x: shift operator o: delta operator.

Figure 4.2: Relative errors for Gaussian spectral density with interpixel distance (a) 0.1 (b) 0.3.

References

[1] P. A. Maragos, R. W. Shafer, and R. M. Mersereau, "Two-dimensional linear prediction and its application to adaptive predictive coding of images," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, pp. 1213–1229, Dec. 1984.

[2] R. L. Kashyap and R. Chellappa, "Estimation and choice of neighbors in spatial-interaction models of images," *IEEE Transactions on Information Theory*, vol. 29, pp. 60–72, Jan. 1983.

[3] R. Vijayan, H. V. Poor, J. B. Moore, and G. C. Goodwin, "A Levinson-type algorithm for modeling fast-sampled data," *IEEE Transactions on Automatic Control*, vol. 36, pp. 314–321, Mar. 1991.

[4] L.-H. Yiin and H. V. Poor, "Linear interpolation models for rapidly-sampled data," in *Proceedings of Conference on Information Sciences and Systems*, pp. 261–266, Mar. 1995.

[5] P. Whittle, "On stationary processes in the plane," *Biometrika*, vol. 41, pp. 434–449, 1954.

[6] V. Heine, "Models for two-dimensional stationary stochastic processes," *Biometrika*, vol. 42, pp. 170–178, 1955.

[7] A. K. Jain, "Partial differential equations and finite-difference methods in image processing, part 1: Image representation," *Journal of Optimization Theory and Applications*, vol. 23, pp. 65–91, Sept. 1977.

[8] A. V. Vecchia, "A general class of models for stationary two-dimensional random processes," *Biometrika*, vol. 72, no. 2, pp. 281–291, 1985.

[9] J. V. Krogmeier and K. S. Arun, "On the recursive computation of interpolators with irregular masks," *IEEE Transactions on Signal Processing*, vol. 44, pp. 1072–1079, May 1996.

[10] F. Jabbari, "Lattice filters for RLS estimation of a delta operator-based model," *IEEE Transactions on Automatic Control*, vol. 36, pp. 869–875, July 1991.

[11] X. Liu and H. Fan, "Unequal-length multichannel δ_b-Levinson and Schur type RLS algorithms," *IEEE Transactions on Signal Processing*, vol. 43, pp. 336–341, Jan. 1995.

[12] A. M. Sequeira and C. W. Therrien, "A new 2-D fast RLS algorithm," in *Proceedings of ICASSP*, (Albuquerque, New Mexico), Apr. 1990.

Connections Between Multidimensional Systems Theory and Algebraic Geometry *

Chris Peterson
University of Notre Dame, Notre Dame, Indiana 46556
e-mail: peterson@math.nd.edu

Joachim Rosenthal
University of Notre Dame, Notre Dame, Indiana 46556
e-mail: Joachim.Rosenthal@nd.edu

Paul A. Weiner
University of Notre Dame, Notre Dame, Indiana 46556
e-mail: pweiner@artin.helios.nd.edu

Abstract

The application of algebraic geometry to 1-D systems theory has led to advances in realization theory (see e.g. [11, 16]), pole placement (see e.g. [3, 18]), and coding theory. As interest in n-D systems theory and n-D coding theory grows, it will again prove beneficial to apply the tools and techniques of algebraic geometry to obtain new results in these areas.

This paper is intended to explore some of the relationships between n-D systems theory and algebraic geometry, indicating correspondences between some of the basic objects in these two fields.

It is our hope that this correspondence of seemingly different research fields will lead to advances in both.

*This work was supported in part by NSF grant DMS-94-00965.

1 Introduction

1.1 Some Basic Notions From Algebraic Geometry

The interested reader may wish to consult [1, 4, 8].

Let $\mathbb{F}$ be an algebraically closed field (e.g., $\mathbb{C}$). Let $R = \mathbb{F}[s_1, ..., s_n]$ be the ring of polynomials in n indeterminates over $\mathbb{F}$. Let $\{f_1, ..., f_k\}$ be a finite set of polynomials from R. The *affine variety* associated to $\{f_1, ..., f_k\}$ is $V(f_1, ..., f_k) = \{(a_1, ..., a_n) \in \mathbb{F}^n : f_i(a_1, ..., a_n) = 0 \text{ for all } i\}$. So in general an affine variety is the common zero set in $\mathbb{F}^n$ of a (finite) set of polynomials from R. A *hypersurface* is a variety determined by a single nonconstant polynomial from R.

It turns out that the collection of affine varieties may be taken to be the closed sets of a topology on $\mathbb{F}^n$, called the *Zariski topology*. So a Zariski closed set is a variety; a Zariski open set is the complement in $\mathbb{F}^n$ of a variety.

A variety V is *irreducible* if it cannot be written as $V = V_1 \cup V_2$ where V_1 and V_2 are both varieties properly contained in V. Any variety V can be written uniquely up to order as $V = V_1 \cup V_2 \cup ... \cup V_k$ where each V_i is an irreducible variety and if $i \neq j$ then V_i is not a subset of V_j. The V_i's are called the irreducible components of V.

Given an irreducible variety V, we define its dimension as follows: Write $V = V_0 \supset V_1 \supset V_2 \supset ... \supset V_d \supset \phi$ where each containment is proper, each V_i is an irreducible variety, and d is as large as possible (such d exists and in fact $d \leq n$). The *dimension* of V is given by $\dim(V) = d$. By convention, $\dim(\phi) = -1$. The dimension of an arbitrary variety is defined to be the maximum dimension of all its irreducible components. For example, $\dim(\text{ point }) = \dim(\text{ finite set }) = 0$; $\dim(\text{ hypersurface }) = n - 1$.

$\mathbb{P}^n$ denotes projective n-space (i.e., the set of 1-dimensional subspaces of $\mathbb{F}^{n+1}$). A point in $\mathbb{P}^n$ thus corresponds to an equivalence class of points in $\mathbb{F}^{n+1} \backslash \{0\}$ where two points are identified if they both lie in the same 1-dimensional subspace of $\mathbb{F}^{n+1}$. $\mathbb{P}^n$ can thus be given *homogeneous coordinates* $[c_0, ..., c_n] \neq [0, ..., 0]$ with $[c_0, ..., c_n] \sim [d_0, ..., d_n]$ if there exists $\lambda \in \mathbb{F} \backslash \{0\}$ such that $\lambda \cdot c_i = d_i$ for $i = 0, ..., n$. A *homogeneous polynomial* is one in which all terms have the same total degree (e.g., x^2y has total degree 3). If G is a homogeneous polynomial in $n + 1$ variables, then the zero-locus of G determines a collection of points in $\mathbb{P}^n$. A *projective variety* in $\mathbb{P}^n$ is the set of common zeros of a finite set of homogeneous polynomials in $n+1$ variables. Note that this is a well-defined notion, for if $f(a_0, ..., a_n) = 0$ then $f(\lambda a_0, ..., \lambda a_n) = 0$ by homogeneity. A *Zariski open set* in $\mathbb{P}^n$ is the set-theoretic complement of a projective variety in $\mathbb{P}^n$ [1]. The *Grassmann manifold*, $\text{Grass}(r, l)$, is the set of all r dimensional subspaces of $\mathbb{F}^l$ [8]. For example, $\text{Grass}(1, n+1)$ is $\mathbb{P}^n$.

1.2 Notation and Terminology

Let $\mathbb{F}$ be a field (e.g., the field $\mathbb{R}$ of real numbers or the field $\mathbb{C}$ of complex numbers). If P is a $k \times l$ matrix of polynomials over $\mathbb{F}$, then P is of *rank* r if r is the largest integer for which P has a nonzero $r \times r$ minor. We will write P^t for the transpose of the matrix P.

For $n \geq 1$, the set of functions from $\mathbb{N}^n$ to $\mathbb{F}^l$, denoted $(\mathbb{F}^l)^{\mathbb{N}^n}$, will be called the

(n-dimensional) sequence space of $\mathbb{F}^l$. A typical element of $(\mathbb{F}^l)^{\mathbb{N}^n}$ is

$$w = w(j_1, ..., j_n) = \begin{bmatrix} w_1(j_1, ..., j_n) \\ \vdots \\ w_l(j_1, ..., j_n) \end{bmatrix}.$$

If $n = 1$, we use j instead of j_1 in this notation.

Note: In what follows, we have written in terms of discrete-time systems, but most of this material applies equally well to the continuous-time case.

2 Review of the 1-D Case

Note: Some of the material in this section is generalized by the material in the next section, and so we have not fully elaborated on it in this section.

Let $R = \mathbb{F}[s]$ be the ring of polynomials in one indeterminate over $\mathbb{F}$. Let $P(s)$ be a $k \times l$ polynomial matrix of rank r with entries in R. Then $P(s)$ can be associated to the following four objects of interest:

(i) A 1-D behavioral system in the sense of Willems [21] via the map between sequence spaces

$$\left(\mathbb{F}^l\right)^{\mathbf{N}} \xrightarrow{P} \left(\mathbb{F}^k\right)^{\mathbf{N}}; \quad w(j) = \begin{bmatrix} w_1(j) \\ \vdots \\ w_l(j) \end{bmatrix} \longmapsto P(\sigma) \cdot w(j)$$

where σ is the left shift operator (so $\sigma^m w_i(j) = w_i(j+m)$). The behavioral system, $\mathcal{B} = \mathcal{B}(P)$ is the kernel of this map.

(ii) A finitely generated R-module M given by $M = R^l/(P^t \cdot R^k) = R^l/\text{colspace}(P^t)$ (here we are considering elements of R^l as column vectors).

(iii) A 'Hermann–Martin' map from a Zariski open set of $\mathbb{P}^1$ to $Grass(r, l)$. The map is defined by

$$[s, q] \longmapsto \text{rowspace}(P(s, q))$$

where $P(s, q)$ is the row-by-row homogenization of $P(s)$. For example, if

$$P(s) = \begin{bmatrix} s & s^2+1 & s^3 \\ s+1 & s^2 & s^2+s \end{bmatrix}$$

then

$$P(s, q) = \begin{bmatrix} sq^2 & s^2q+q^3 & s^3 \\ sq+q^2 & s^2 & s^2+sq \end{bmatrix}.$$

Specifically the $r \times r$ minors of P(s,q) determine a variety in $\mathbb{P}^1$. The points of this variety are the points of $\mathbb{P}^1$ at which P(s,q) drops rank. The Zariski open set mentioned above is the complement of this variety. Hermann and Martin [13] recognized this map as being of fundamental interest. They relate this map to the McMillan degree and the observability indices of the behavioral system given in (i).

Also of interest here is if $P(s)$ is of constant rank r (i.e., of rank r for any particular value of s), and if also P_∞ (:=matrix of coefficients of highest powers in each row of $P(s)$) is of rank r, then the Zariski-open set mentioned above is all of $\mathbb{P}^1$. There is in fact a one-to-one correspondence between holomorphic maps from $\mathbb{P}^1 \longrightarrow \text{Grass}(r, l)$ and matrices $P(s)$ such that $P(s)$ is of constant rank r and is in row proper form (i.e., P_∞ is also of rank r). Systems theoretically such $P(s)$ define controllable systems.

(iv) A coherent sheaf on $\mathbb{P}^1$, together with a map between two coherent sheaves, each of which splits as a sum of line bundles [9].

As mentioned previously, the interconnection of these different research areas has led to advances in realization theory and pole placement theory. For a recent application to interpolation theory see [15].

3 The n-D Case

Multidimensional systems are by definition under-determined systems of linear partial differential (or difference) equations. The class of multidimensional systems are widely studied both in systems theory and in coding theory (see e.g. [2, 5, 14, 19]) and there is even a specialized journal (titled *Multidimensional Systems and Signal Processing*) on this area of research. It has been recognized that algebraic methods, in particular Gröbner basis techniques (see e.g. [14]) are important in the study of these systems. The application of algebraic geometry to 1-D systems theory has led to advances in realization theory (see e.g. [11, 16]), pole placement (see e.g. [3, 18]), and coding theory (see e.g. [7]). As interest in n-D systems theory and n-D coding theory grows, it will again prove beneficial to apply the tools and techniques of algebraic geometry to obtain new results in these areas. We are aiming at a Hermann-Martin identification of the class of multidimensional systems and such an identification would allow one to carry over techniques from algebraic geometry into the field of multidimensional systems. It is in particular our hope to develop a realization theory for multidimensional systems and to develop a theory of feedback interconnections. In order to achieve such goals it will be necessary to identify a multidimensional system with an 'algebraic object'.

It is worth noting that one important difference between the 1-D case and the n-D case is that a polynomial ring in one variable over a field is a principal ideal domain, while a polynomial ring in two or more variables is not. This difference complicates matters considerably in the n-D case.

Let $R = \mathbb{F}[s_1, ..., s_n]$ be the ring of polynomials in n indeterminates over $\mathbb{F}$. Let $P = P(s_1, ..., s_n)$ be a $k \times l$ polynomial matrix of rank r with entries in R. As in the 1-D case, P can be viewed in four ways:

(i) An n-D behavioral system via the map between sequence spaces

$$\left(\mathbb{F}^l\right)^{\mathbb{N}^n} \longrightarrow \left(\mathbb{F}^k\right)^{\mathbb{N}^n}$$

$$w(j_1,...,j_n) = \begin{bmatrix} w_1(j_1,...j_n) \\ \vdots \\ w_l(j_1,...,j_n) \end{bmatrix} \longmapsto P(\sigma_1,...,\sigma_n)\cdot w(j_1,...j_n)$$

where σ_m is the left shift operator acing on the m-th coordinate. Again the behavior, denoted $\mathcal{B} = \mathcal{B}(P)$, is the kernel of this map.

(ii) A finitely generated R-module M given by $M = R^l/(P^t \cdot R^k) = R^l/\text{colspace}(P^t)$ (here we are considering elements of R^l as column vectors).

In a major paper Oberst [14] establishes a categorical duality (i.e. a one-to-one correspondence with special properties) between R-modules of the form $R^l/\text{colspace}(Q)$ where Q is a matrix with l rows (here we consider the elements of R^l as column vectors). and behavioral systems in $\left(\mathbb{F}^l\right)^{\mathbb{N}^n}$ as introduced above. We describe this duality more fully in the next section.

Using this duality it is possible to identify a multidimensional system with a module. Alternatively we can also view a multidimensional system in two geometric ways:

(iii) As a map from a Zariski open set of $\mathbb{P}^n$ to $\text{Grass}(r,l)$. The map is defined by

$$[s_1...,s_n,q] \longmapsto \text{rowspace}(P(s_1,...,s_n,q))$$

where $P(s_1,...,s_n,q)$ is the row-by-row homogenization of $P(s_1,...,s_n)$. (Again the open set is the complement of the variety defined by the $r \times r$ minors of P).

(iv) As a coherent sheaf on $\mathbb{P}^n$ together with a map between two coherent sheaves, each of which splits as a sum of line bundles. The sheaves and map arise from homogenizing P and sheafifying the exact sequence (4.2) below with P replaced by its homogenization. This identifies a geometric object (a coherent sheaf) with an algebraic object (the module in (ii)), and allows us to apply algebro-geometric techniques to objects originating in n-D systems theory.

4 The Module-System Duality of Oberst

At this point it is worth mentioning some of the results of Oberst [14] as they are relevant to items (i) and (ii) in each of the two preceding sections. In the rest of this section, page numbers in parentheses refer to this article by Oberst. We maintain the notation of the previous section.

Regarding item (i) in the previous section, Oberst (pp. 61-62) gives an intrinsic characterization of n-D behavioral systems. He proves that a subset $\mathcal{B} \subseteq \left(\mathbb{F}^l\right)^{\mathbb{N}^n}$ is a behavior if and only if $\mathcal{B}$ is linear, left shift invariant, and complete (under a suitable topology). This generalizes a result of Willems in the 1-D case [20].

Now we give a description of the duality mentioned in item (ii) of the previous section. See Oberst (pp. 3, 34).

A behavioral system $\mathcal{B}$ is induced by some polynomial matrix P. That is, $\mathcal{B} = \mathcal{B}(P)$. Say P is a $k \times l$ matrix. Then $P^t : R^k \longrightarrow R^l$ via matrix multiplication. The cokernel of this map is the R-module $M = \text{coker}(P^t) = R^l/(P^t \cdot R^k)$. So $\mathcal{B}$, the kernel of the map P on the system side, corresponds (is dual to) M, the cokernel of the map P^t on the module side. The following exact sequences illustrate the situation.

System side:

$$0 \longrightarrow \mathcal{B}(P) \longrightarrow \left(\mathbb{F}^l\right)^{\mathbb{N}^n} \xrightarrow{P} \left(\mathbb{F}^k\right)^{\mathbb{N}^n} \tag{4.1}$$

Module side:

$$R^k \xrightarrow{P^t} R^l \longrightarrow R^l/(P^t \cdot R^k) \longrightarrow 0 \tag{4.2}$$

One immediate consequence of this duality is that if P' is a $k' \times l$ matrix, then $\mathcal{B}(P) = \mathcal{B}(P')$ if and only if there are matrices A and A' of respective sizes $k' \times k$ and $k \times k'$ such that $A \cdot P = P'$ and $A' \cdot P' = P$. This follows because from the duality, $\mathcal{B}(P) = \mathcal{B}(P')$ if and only if rowspace(P) = rowspace(P').

We note that it is possible to have matrices P and P' for which colspace(P^t) and colspace(P'^t) are isomorphic as R-submodules of R^l but which correspond to different behaviors under the duality. That is the duality depends not only on the isomorphism class of colspace(P^t) but on the actual column space itself.

Example 4.1 We are in the 1-D case. Let

$$P = \begin{bmatrix} s & s+1 \end{bmatrix} \qquad P' = \begin{bmatrix} 1 & s+1 \end{bmatrix}$$

The map

$$\begin{array}{ccc} \phi : R^2 & \longrightarrow & R^2 \\ \begin{bmatrix} a \\ b \end{bmatrix} & \longmapsto & \begin{bmatrix} b-a \\ b \end{bmatrix} \end{array}$$

is an isomorphism of R^2 to itself that takes colspace(P^t) onto colspace(P'^t). However the behaviors $\mathcal{B}(P) \neq \mathcal{B}(P')$. For instance

$$\begin{bmatrix} 1 & 1 & 0 & 0 & 0 & \cdots \\ -1 & 0 & 0 & 0 & 0 & \cdots \end{bmatrix}$$

is in $\mathcal{B}(P)$ but not in $\mathcal{B}(P')$.

Sometimes it is also natural to consider the correspondence between $\mathcal{B}(P)$ and the R-module rowspace(P). In fact rowspace(P) carries exactly the same information as $R^l/$colspace(P^t). So the above duality automatically gives a one-to-one correspondence between R-submodules of R^l (any one of which is generated by a finite number of elements of R^l and these elements form the rows of a matrix P whose row space is the particular R-submodule) and behavioral systems in $\left(\mathbb{F}^l\right)^{\mathbb{N}^n}$. This one-to-one correspondence can also be expressed as a duality using a certain bilinear form on $R^l \times \left(\mathbb{F}^l\right)^{\mathbb{N}^n}$. For this we

will identify $\left(\mathbb{F}^l\right)^{\mathbb{N}}$ with the ring of formal power series $\mathbb{F}[[s_1, \ldots, s_n]]$. An element $w(s)$ of the sequence space $\left(\mathbb{F}^l\right)^{\mathbb{N}^n}$ is then a formal sum

$$w(s) = \sum_{i \in \mathbb{N}^n} w_i s^i = \sum_{(i_1,\ldots,i_n) \in \mathbb{N}^n} w_{(i_1,\ldots,i_n)} s_1^{i_1} \cdots s_n^{i_n}.$$

Using this identification we define a bilinear form already used by Macaulay [12] (compare also with [17]):

$$\begin{aligned} <,>: \quad \mathbb{F}[[s_1, \ldots, s_n]] \times \mathbb{F}[s_1, \ldots, s_n] &\longrightarrow \mathbb{F} \\ (w(s), p(s)) &\longmapsto \sum_{i \in \mathbb{N}^n} w_i p_i \end{aligned} \tag{4.3}$$

where $w_i p_i$ is the standard inner product on two l-component vectors. Under this bilinear form behavioral systems in $\left(\mathbb{F}^l\right)^{\mathbb{N}^n}$ correspond to submodules of R^l. We omit the details in this paper.

5 Primeness for Polynomial Matrices

Fornasini and Valcher relate several notions of primeness of a polynomial matrix to systems theoretic concepts such as observability and extendability [6]. In this section we relate these definitions of primeness to concepts of algebraic geometry given in section 1.1 above.

We assume here that $\mathbb{F}$ is algebraically closed. The reader may think of $\mathbb{F} = \mathbb{C}$.

Let $P = P(s_1, ..., s_n)$ be a $k \times l$ polynomial matrix over $R = \mathbb{F}[s_1, ..., s_n]$. Assume P is of full rank (i.e., P has a nonzero full-size minor). Let J be the ideal of R generated by the full-size minors of P. Let $V(J) = \{(a_1, ..., a_n) \in \mathbb{F}^n : f(a_1, ..., a_n) = 0 \text{ for all } f \in J\}$. That is $V(J)$ is the variety of J.

Following Fornasini and Valcher we have the following definition.

Definition 5.1 P is

(i) zero prime if $J = R$ (i.e., the constant polynomial 1 can be written as a polynomial combination of the full-size minors of P).

(ii) variety prime if for each $i = 1, ..., n$, J contains a nonzero polynomial in the variable s_i alone.

(iii) minor prime if there is no nonconstant polynomial that divides every element of J.

Proposition 5.2 *(i) P is zero prime if and only if $V(J) = \phi$.*

(ii) P is variety prime if and only if $V(J)$ is a finite set of points in $\mathbb{F}^n$.

(iii) P is minor prime if and only if $V(J)$ contains no hypersurface in $\mathbb{F}^n$ (a hypersurface is the set of zeros of a single polynomial in R).

Proof:

(i) follows immediately from the weak form of Hilbert's Nullstellensatz [4].

(ii) Suppose P is variety prime. Choose $f_i \in J$ such that f_i depends only on s_i. Each f_i has only finitely many, say k_i, zeros in $\mathbb{F}$. If $(a_1, ..., a_n) \in V(J)$ then each a_i must be one of the k_i zeros of f_i. There are only $k_1 \cdot ... \cdot k_n$ such $(a_1, ..., a_n)$. So $V(J)$ has at most $k_1 \cdot ... \cdot k_n$ elements.
Conversely suppose $V(J) = \{x^{(1)}, ..., x^{(t)}\}$ is finite. Say $x^{(j)} = (x_1^{(j)}, ..., x_n^{(j)})$. Put $f_i = (s_i - x_i^{(1)})(s_i - x_i^{(2)})...(s_i - x_i^{(t)})$. By construction, f_i vanishes at each point of $V(J)$ and hence, by Hilbert's Nullstellensatz [4], some power of f_i is in J. But any power of f_i depends only on the variable s_i. So $V(J)$ is variety prime.

(iii) immediate from the definition.

□

From the discussion in Section 1, the above proposition can be recast as:

P is zero prime if and only if $\dim(V(J)) = -1$;

P is variety prime if and only if $\dim(V(J)) \leq 0$;

P is minor prime if and only if $\dim(V(J)) \leq n - 2$.

So we can generalize definition 4.1:

Definition 5.3 *P is $m-$prime $(0 \leq m \leq n)$ if* $\dim(V(J)) \leq m - 1$. *P is strictly $m-$prime if* $\dim(V(J)) = m - 1$.

So zero prime is $0-$prime; variety prime is $1-$prime; and minor prime is $(n-1)-$prime.

6 Research Questions

1. The associated Hermann-Martin map as we described it above depends a priori on the polynomial matrix P, which represents the multidimensional system. It raises the question if every multidimensional system induces in a 'canonical way' a Hermann–Martin map, i.e., a map only dependent on the multidimensional behavior.

2. We are interested in a realization theory for n-D systems. Some work has been done in this area by Oberst [14]. What we have in mind is based on the following observation: In the 1-D context the generalized first order representations, i.e., realizations (see [10]) can be viewed as the intertwining maps of the free resolution of the associated quotient sheaf (see [16]). Also in the multidimensional situation we have an associated quotient sheaf which is in this case not a short exact sequence in general. We are wondering if we can view this free resolution as a realization.

3. Willems [22] did start to develop a theory of feedback interconnections for 'abstract behaviors'. We would like to study such interconnections in the more concrete setting of multidimensional systems and extending in this way some results obtained in the context of 1-D systems.

4. So far we did describe research projects where tools from algebraic geometry were applied to systems theory. There is actually an interesting twist which goes into the 'other direction'. Indeed some matrix representations used in the area of multidimensional systems might be useful in the construction of non-trivial low rank vector bundles on projective spaces. Such constructions are of ample importance in algebraic geometry and this is underlined by the fact that most low rank nontrivial bundles on $\mathbb{P}^2$ and $\mathbb{P}^3$ have special names.

7 Conclusion

We intend this paper as a beginning point for our research and, we hope, the research of others into the interplay between systems theory and algebraic geometry. As indicated above, we believe that the study of this interplay should prove fruitful indeed for both areas.

References

[1] S. Abhyankar. *Algebraic Geometry for Scientists and Engineers.* American Mathematical Society, Providence, R.I., 1990.

[2] N.K. Bose. *Multidimensional Systems Theory.* Reidel, Dordrecht, 1985.

[3] R. W. Brockett and C. I. Byrnes. Multivariable Nyquist criteria, root loci and pole placement: A geometric viewpoint. *IEEE Trans. Automat. Control*, AC-26:271–284, 1981.

[4] D. Cox, J. Little, and D. O. O'Shea. *Ideals, Varieties and Algorithms.* Undergraduate Text in Mathematics. Springer Verlag, New York, 1992.

[5] E. Fornasini and M.E. Valcher. Algebraic aspects of 2D convolutional codes. *IEEE Trans. Inform. Theory*, IT-40(4):1068–1082, 1994.

[6] E. Fornasini and M.E. Valcher. nd polynomial matrices with applications to multidimensional signal analysis. *Multidimensional Systems and Signal Processing*, To appear, 1996.

[7] V. D. Goppa. *Geometry and Codes.* Kluwer Academic Publisher, 1988.

[8] J. Harris. *Algebraic Geometry, A First Course.* Graduate Text in Mathematics. Springer Verlag, New York - Berlin, 1992.

[9] G. R. Kempf. *Algebraic Varieties.* London Mathematical Society Lecture Note Series # 172. Cambridge University Press, 1993.

[10] M. Kuijper. *First-Order Representations of Linear Systems.* Birkhäuser, Boston, 1994.

[11] V. G. Lomadze. Finite-dimensional time-invariant linear dynamical systems: Algebraic theory. *Acta Appl. Math*, 19:149–201, 1990.

[12] F.S. Macaulay. *The Algebraic Theory of Modular Systems.* Cambridge University Press, Cambridge, 1916.

[13] C. F. Martin and R. Hermann. Applications of algebraic geometry to system theory: The McMillan degree and Kronecker indices as topological and holomorphic invariants. *SIAM J. Control Optim.*, 16:743–755, 1978.

[14] U. Oberst. Multidimensional constant linear systems. *Acta Appl. Math*, 20:1–175, 1990.

[15] M. S. Ravi. Geometric methods in rational interpolation theory. *Linear Algebra Appl.*, 1996. To appear.

[16] M. S. Ravi and J. Rosenthal. A general realization theory for higher order linear differential equations. *Systems & Control Letters*, 25(5):351–360, 1995.

[17] J. Rosenthal, J. M. Schumacher, and E.V. York. On behaviors and convolutional codes. *IEEE Trans. Inform. Theory*, 42(6), 1996. To appear.

[18] J. Rosenthal and X. Wang. Output feedback pole placement with dynamic compensators. *IEEE Trans. Automat. Contr.*, 41(6):830–843, 1996.

[19] M. E. Valcher. *Modellistica ed Analisi dei Sistemi 2D con Applicazioni alla Codifica Convoluzionale.* PhD thesis, University of Padova, 1995.

[20] J. C. Willems. From time series to linear system. Part I: Finite dimensional linear time invariant systems. *Automatica*, 22:561–580, 1986.

[21] J. C. Willems. Paradigms and puzzles in the theory of dynamical systems. *IEEE Trans. Automat. Control*, AC-36(3):259–294, 1991.

[22] J. C. Willems. On interconnections, control and feedback. Preprint, submitted to IEEE AC, July 1994.

Calibration-Free Vision Based Control of a Robotic Manipulator*

Bijoy K. Ghosh, Di Xiao, Ning Xi and T. J. Tarn
Department of Systems Science and Mathematics
Washington University
One Brookings Drive
Saint Louis, MO 63130-4899, USA

Abstract

In this paper we introduce a new calibration-free approach to the problem of visual manipulation of a part placed randomly on a rotating turntable. The robot end effector and the camera are both assumed to be uncalibrated initially and our approach emphasizes obtaining geometric cues from the environment visually in order to obtain the position and orientation information for the part and the end effector. Calibration is not performed separately but as a part of the overall estimation, planning and tracking scheme. The implementation is recursive and distinguishes from many classical approaches in the literature wherein calibration is to be performed prior to tracking and manipulation.

1 Introduction

Control of robot manipulators with vision in the feedback loop has a long history both in Robotics and in the Machine Vision literature. The specific area of research that has emerged out of this goes by the name of 'visual servoing' [18], wherein the emphasis is to visually locate the position and orientation of an object while controlling a robot manipulator to grasp and manipulate the object. If the object is moving, this process has to be implemented iteratively until the task has been accomplished, leading to real time vision based feedback problems that has been studied in [1]-[5].

Guided by the problems and prospects of visual servoing, we propose in this paper, a new visually guided planning and control of a robot manipulator for precise positioning and tracking in an uncertain and dynamic environment. The proposed scheme is robust against precise position of the camera and utilizes a scheme called *virtual rotation* wherein the observed image is transformed to what it would appear if the camera has been pointed vertically down. Virtual rotation scheme differs from prior approaches in the literature [15] in the sense that there is no actual camera motion. This leads to a relatively simple hardware for implementation of the scheme. Additionally the proposed scheme is also robust against precise positioning of the robot manipulator. This is achieved by a new calibration-free technique which we propose in this paper and has been explained in section 3. Roughly speaking, the technique involves observing feature points on the end effector using a CCD camera and estimating the relative position of the base of the robot with respect to a standard coordinate system on the work-cell.

Specifically in this paper we report further results on our ongoing investigation on visually controlled manipulation of parts on a rotating turntable with the aid of a single CCD camera. Earlier results have been reported in [6], [7] and somewhat complete details have been provided in [8]. Our approach differs from the traditional visual servoing as has been summarized in [1]-[5] in the following sense.

1. Real time part localization does not entirely depend on the CCD camera. Information from encoders attached to the rotating turntable is also used in a new multisensor fusion based scheme.

2. The tracking control is derived using a new event-based strategy. The event based scheme has been originally reported by Tarn, Bejczy and Xi [9] wherein the target point is assumed to be a fixed point in space. Our approach differs from [9] in the sense that we propose to track a moving target, since the part on the turntable is constantly moving.

*Partially supported by DOE under grant number DE-FG02-90ER14140, NSF grant number IRI-9106317, CAD-9404949 and by Sandia National Laboratories contract number AC 3752-C. This paper has been submitted for IROS'96.

3. In order to track and grasp a part on a rotating turn-table, it is important to be able to estimate the position and orientation of the moving part, in front of a CCD camera. We propose a new scheme where the computational burdon of estimation is reduced via multisensor fusion. Furthermore the scheme does not rely on priori knowledge of the exact position and orientation of the camera.

4. In general, the task of manipulation in a remote environment would be performed by a robotic manipulator, that would be mounted on a mobile platform. Because of its mobility, the camera coordinate and the robot coordinate would not maintain a fixed relationship, and in general it would be impractical to calibrate the robot every time it is to be used. The new calibration free scheme, proposed in this paper, utilizes a set of apriori chosen points on the robotic end-effector to compute the coordinate transformation for the robotic manipulator. The scheme is calibration-free because the transformation is computed on line along with the position and orientation of the robotic end effector and no separate calibration step is needed.

We would also like to point out that the estimation scheme presented in this paper differs considerably from standard results in motion and structure estimation from feature correspondences (see [12], [13], [14] and [20]). Although our approach is feature based, we do not propose to visually estimate structure and motion. In fact, on one hand, we assume that the motion of the moving part is obtained by fusing visual information from the camera with readings from an encoder attached to the rotating turntable. On the other hand, the structure information in terms of the orientation of the end effector, is measured in terms of the base coordinate frame of the robot using the encoders attached to the robot. The main issue addressed in this paper is "how to fuse information obtained by various sensors in different coordinate frames, without explicitly having to calibrate." The main conclusion is that the procedure of sensor fusion described in this paper leads to an unique calibration free algorithm which is reliable and has been tested in simulation and via experiments.

We consider a manufacturing workcell as shown in Figure 1. The workcell is equipped with a rotating turn table, a robotic manipulator and a machine vision system with a CCD camera. We assume that a part with a known geometric shape has been placed randomly on the rotating turntable. The position and orientation of the part is therefore assumed to be apriori unknown. Furthermore we also assume that the base of the robotic manipulator is on a mobile platform because of which the exact position and orientation of the end-effector in the camera co-ordinate is not known and in any case is changing in time as the mobile platform moves.

Our basic problem is to obtain the position, orientation and the motion parameters of the moving part in the robot co-ordinate system with the aid of a single camera fixed rigidly in space while the part moves on the turntable and the platform moves on the surface. The motion of the platform is assumed to be entirely unknown while the motion of the part is assumed to be circular with an unknown angular velocity.

One of the main ingredient of this paper is the calibration-free process utilizing which we compute the position and orientation of the end-effector in the camera co-ordinate system. The calibration-free process is implemented by assuming that a set of points is observed on the *Robot End Effector*. In general, the location of these points are assumed to be fixed in the coordinate frame of the robot. However since the base and the end effector of the robot moves in time, the coordinates of the points have to be determined visually with the aid of a CCD camera. The procedure has been detailed in the next section.

To end this section we make this following important remark. The calibration-free process proposed in this paper is essentially to estimate the position and orientation of the end effector with respect to the camera coordinate frame. The estimation scheme proposed differs from the traditional CCD camera based estimation in the sense that we use data from both the image plane of the camera and from the encoders of the robot. In particular, the robot encoders provide distance and angle information that are invariant to rotation with respect to the vertical z-axis. Thus the scheme utilizes fusion between robot encoders and the image plane of the camera. In this sense the problem is different from that reported in [13], [14] and [20].

2 Details of the Calibration Free Scheme

Recall from section 1 the assumption that the relative position of the camera and the end-effector w.r.t. the turntable are both unknown. We propose to rotate the camera virtually and obtain the image from the viewpoint where the camera is pointed vertically down. This procedure has been detailed in

our earlier work [6]–[8]. The main point is that the algorithm can be implemented no matter what the original viewpoint of the camera is.

In this section, we propose to describe in detail the procedure utilizing which we can compute the relative transformation between the co-ordinate systems attached to the base of the robot and the rotating turntable. Of course it is clear that for the purpose of implementing the control algorithm in order to track and grasp the rotating part, it is necessary to compute such a co-ordinate transformation.

The algorithm we propose is feature point based wherein we choose a set of points on the end effector of the robot. The co-ordinates of the points are assumed known in the robot-co-ordinate frame and can readily be read off from two robot-encoders. The perspective projection of the points in the image plane are also assumed known (upto a possible permutation of the points) and can readily be computed from the image. Note in particular that we do not assume correspondence of individual points but only require correspondence of the set of points (i.e. correspondence upto permutation). Thus individual feature point recognition is not necessary.

The problem is to compute the co-ordinates of the feature points on the end-effector with respect to the co-ordinate frame attached to the rotating turntable.

Clearly, if a set of points are known with respect to the robot and the turntable coordinate frames, the required co-ordinate transformation can be readily computed.

Three subcases are now presented for the solution to the above problem of computing the end-effector co-ordinates.

2.1 Algorithm based on 2 points (height of the turntable known)

Here we assume that the height of the turntable is known with respect to the base of the robot. Furthermore we assume that the x-y planes of the turntable and the robot co-ordinates are parallel. We place two points on the end-effector and observe the points using an uncalibrated camera and show that the co-ordinates of the points can be computed uniquely in the turntable co-ordinate frame.

Theoretical details of this algorithm has been sketched out in [10]. Subsequently this algorithm has been experimentally verified and these results are reported in this paper.

2.2 Algorithm based on 2 points (height of the turntable unknown)

Here we relax the assumption that the height of the turntable is known with respect to the base of the robot. We continue to assume that the x-y planes of the turntable and the robot co-ordinates are parallel.

Two points are placed on the robot end effector and are observed by an uncalibrated camera. Using the robot encoders, we measure the distance between the two points and the angle that the line segment joining the two points make with respect to the vertical axis.

The main result we obtain is that the unknown height of the turntable can be computed in general upto a pair of choices. In many special cases the choices is unique and these cases can be characterized.

2.3 Algorithm based on 3 points (height of the turntable unknown)

Since in 2.2 we do not get a unique solution in general, we are forced to extend our algorithm to 3 points. Let us make assumptions as in 2.2. We place 3-feature points on the end effector and these points are all observed by an uncalibrated camera. Furthermore only the collection of the images of the line feature points are required. Correspondence of individual points does require. Using the robot encoders, we measure distance between every pairs of feature points and the angle each of the line segments joining two points make with respect to the vertical axis.

The main result we show that generically we can compute the following uniquely

1. The 2D-3D correspondence of each of the 3 feature points.
2. The height of the turntable.
3. The co-ordinates of the feature points with respect to the turntable co-ordinate frame.

The procedure has been described in section 3. We would like to remark that 2.3 describes a 3-point algorithm which is different from that described in the photogrametric literature (see for example [13],[14]). In fact we obtain unique solution under a slightly stronger hypothesis viz co-ordinates of the feature

points are known with respect to the robot co-ordinates. Three point algorithms described in the literature [13],[14] is under a different hypotheses leading to a non-unique solution. Furthermore, questions about computing height of the turntable and co-ordinate transformation has not been considered to the literature.

3 Some Mathematical Comments

In this section, we justify some of the comments made in sections 2.2 and 2.3. Let us assume that (x_c, y_c, z_c) be a co-ordinate system attached to the camera. Let us assume that there are points, suitable chosen, on the end effector with co-ordinates (x_i, y_i, z_i), $i = 1, 2, ...$, with respect to the camera. We shall assume that the total number of points are either two or three. Starting the discussion with three points we show that under a suitable generic assumption on the position of the points its co-ordinates are computable uniquely w.r.t. the turntable co-ordinates. In fact the problem reduces to solving a pair of quadratic polynomial equations in one variable with the property that the polynomials have a common real root. Subsequently, we remark, that if instead of three points we only have two points observed on the end effector then the position and orientation can be computed up to a pair of choices. Mathematically the problem reduces to solving a single quadratic polynomial equation in one variable with the property that the polynomial has real roots. Hence there is always a pair of real roots. The details of the procedure are described as follows.

Note first of all that the camera observes points via perspective projection. In fact each of the point is observed as the line which these points make with respect to the origin. Stated differently, each of the points satisfy an equation of the type

$$\begin{pmatrix} a_1 & a_2 & a_3 \\ a_4 & a_5 & a_6 \end{pmatrix} \begin{pmatrix} x_1 \\ y_1 \\ z_1 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{3.1}$$

$$\begin{pmatrix} a_4 & a_5 & a_6 \\ a_7 & a_8 & a_9 \end{pmatrix} \begin{pmatrix} x_2 \\ y_2 \\ z_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{3.2}$$

$$\begin{pmatrix} a_7 & a_8 & a_9 \\ a_1 & a_2 & a_3 \end{pmatrix} \begin{pmatrix} x_3 \\ y_3 \\ z_3 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{3.3}$$

Here we assume that the camera observes three points on the end effector. Let us further assume that the robot has it own co-ordinate system with respect to which the magnitude of the vectors

$$\begin{aligned} &(x_1 - x_2, y_1 - y_2, z_1 - z_2), \\ &(x_2 - x_3, y_2 - y_3, z_2 - z_3), \\ &(x_3 - x_1, y_3 - y_1, z_3 - z_1) \end{aligned} \tag{3.4}$$

are known. Each of the above three vectors make an angle $\theta_1, \theta_2, \theta_3$ respectively with respect to the z co-ordinate of the robot, which we shall assume to be known. Thus without any loss of generality we may assume that

$$\begin{aligned} z_1 - z_2 &= \alpha \\ z_2 - z_3 &= \beta \end{aligned} \tag{3.5}$$

where α and β are known from the robot encoder, provided we assume that the z co-ordinates of the robot and camera are parallel. The equations (3.1)-(3.5) constitute a set of eight linear equations in 9 variables which can in general be solved in terms of a single parameter, say z_3. Note also that in order to set up the eight linear equations, information between camera and the robot encoders needs to be fused. Furthermore since the magnitude of the vectors

$$\begin{aligned} &(x_1 - x_3, y_1 - y_3, z_1 - z_3) \\ &(x_2 - x_3, y_2 - y_3, z_2 - z_3) \end{aligned}$$

are assumed known in general, one obtains two quadratic equations additionally in z_3 given by

$$\begin{aligned} \alpha_1 + \alpha_2 z_3 + \alpha_3 z_3^2 &= 0 \\ \beta_1 + \beta_2 z_3 + \beta_3 z_3^2 &= 0 \end{aligned} \tag{3.6}$$

where $\alpha_1, \alpha_2, \alpha_3, \beta_1, \beta_2, \beta_3$ are functions of α, β and $a_1, a_2, \ldots a_9$. Of course the two quadratic equations (3.6) have one solution which is the true solution and generically we argue that the solution is unique.

For computational reasons, or perhaps if three points on the end-effector are not visible, then one has to restrict the computation to only two points. In this case it can be analogously argued that one would obtain a single quadratic equation in z_3 given by only the first equation in (3.6). Since there is one real solution to the quadratic polynomial equation, it follows that there is a pair of real solutions.

4 Multisensor Based Visual Sensing

The purpose of this section is to compute position, orientation and velocity of the part placed on the moving turntable. We shall assume that the part on the turntable has a rectangular cross section and is represented by four corner points. Let us also assume that the orientation of the part is to be measured with respect to an axis on the turntable which is represented by a pair of additional points. Thus there is a total of six points which fixes the orientation of the chosen part on the turntable. We make an important assumption that the height of the part is negligible so that all six points are assumed to be on the plane.

Of course the six points are observed with the aid of a CCD camera which projects these points on the image plane via perspective projection. Such a projection distorts the orientation information unless the camera looks straight down on the turntable.

Via image processing it is possible to derive the parameters that would rotate the camera from an arbitrary initial position to the new position where the camera looks straight down in the turntable. This algorithm would involve some of the ideas originally introduced by Kanatani [11], and developed subsequently in our earlier papers [6],[7],[8] and [10]. For lack of space, we do not go into details here.

5 Robot Planning and Control

In this section we discuss the problem of robot motion planning and control involved in performing tasks such as tracking and grasping a moving part. The problem is how to make a plan for a robot tracking a target whose position and velocity are measured by the sensing schemes discussed in Section 2. The concept of *parallel tracking* is introduced. In this concept an artificial error correction term is added to the position of the target to form a desired motion for the robot. When the artificial error correction term is carefully planned, it guarantees time optimal or robust robot motion with existing bounded control.

The dynamic model of a robot arm is given by

$$D(q)\ddot{q} + C(q, \dot{q}) + G(q) = \tau \tag{5.1}$$

where q, $D(q)$, $C(q, \dot{q})$, $G(q)$ and τ are, separately, the joint angle vector, the inertia matrix, the load related to centripetal and coriolis forces, the load related to gravity, and the joint torque vector. The joint torque has to satisfy the following constraints

$$\tau_{imin}(q, \dot{q}) \leq \tau_i \leq \tau_{imax}(q, \dot{q}) \quad i = 1, 2, \ldots m \tag{5.2}$$

where m is the number of joints. The output is given by $Y = H(q) = (X, \theta)^t$ where X and θ represent the position and orientation of robot end-effector.

The well known nonlinear feedback control law [39] is given by

$$\begin{aligned} \tau = DJ^{-1}(q)(A_d \quad & + K_v(V_d - \dot{Y}) + K_p(Y_d - Y) \\ & - \dot{J}(q)\dot{q}) + C + G \end{aligned} \tag{5.3}$$

where $J(q)$ is the Jacobian of $H(q)$ with respect to q; K_v and K_p, the velocity and position feedback gains; A_d, V_d and Y_d, the desired linear and angular acceleration, velocity and position respectively. It can be shown that if A_d, V_d and Y_d satisfy the constraints $\dot{Y}_d = V_d$ and $\dot{V}_d = A_d$, then, by proper choice of K_v and K_p, the closed-loop system can be made stable and the tracking error will vanish asymptotically provided the induced control or joint torque is within the limit specified by (3.2). The problem of robot motion planning considered here is to design A_d, V_d and Y_d in order to track the motion of the part.

The robot tracking problem is described as follows. Assume that the position $Y_p(t)$ and velocity $V_p(t)$ of a target has been given, obtain a desired trajectory with the property that if the desired trajectory is

followed by the robot end effector, the position error between the robot and the target vanishes. Given the measured position $Y_p(t)$ and velocity $V_p(t)$ of a target, design a desired motion for the robot to track the target, that is to make the positional error between the robot and the target vanish.

An obvious tracking plan would be

$$\begin{cases} Y_d = Y_p(t) \\ V_d = V_p(t) \\ A_d = V_d \end{cases} \tag{5.4}$$

which is to let the desired motion to be the motion of the target. However, when such a plan is used to command the motion of a robot, the required control will be large when the error between Y_d, the position of the part, and Y, the actual position of the robot, is large.

In order to keep the control within acceptable limits, a modified tracking plan is proposed in the form of

$$\begin{cases} Y_d = Y_p(t) + Y_e(t) \\ V_d = \dot{Y}_d = V_p(t) + \dot{Y}_e(t) \\ A_d = \dot{V}_d = A_p(t) + \ddot{Y}_e(t) \end{cases} \tag{5.5}$$

where Y_e is an artificial error correction term that is set to $Y(0) - Y_p(0)$ at the time tracking starts and is gradually reduced to zero according to a plan that is designed based on the bounded control. This plan commands the robot to move at the same speed in parallel to the target while the positional error is gradually reduced according to a plan. This strategy is called parallel tracking in this paper.

6 Simulation and Experimental Results

As has been described in the previous sections, the algorithm proposed for calibration free estimation and robot planning and control has been verified experimentally (see [6] and [7] for details). In this section, we shall focus on the proposed calibration-free schemes for the estimation of the position of the robot. Three subcases introduced in section 2 are now discussed.

As an illustration of the subcase introduced in section 2.1, we conducted an experiment where a robot manipulator (PUMA 560) successfully picked up a part randomly placed on a rotating turntable of diameter 0.9 m, with the aid of a vision system. The vision system consists of a CCD camera with low image resolution of 256 × 256 and the Intelledex Vision processor based on a 16 MHz Intel 80386 CPU. It interfaces to the host computer, a SGI 4D/340 VGX. Image information (pixels) are sent to SGI by a parallel interface. The robot is controlled by UMC controller that also interfaces to SGI through memory mapping. Several markers were used to simplify the image processing.

In the experiment, the positions of camera and the turntable are both unknown in the base frame of the robot. The experimental results are shown in Fig. 2. In Fig. 2, the solid lines denote actual values and the doted lines give the estimated values. The actual position of the part and its estimation provided by the sensor fusion scheme are given in Fig. 2 (a) and (b). Fig. 2 (c) presents the actual and estimated orinetations of the part. It can be seen that the estimation errors are very small and results positively support our scheme.

We now proceed to illustrate the schemes introduced in sections 2.2 and 2.3 via simulation. For the case illustrated in section 2.2, we assume that two points on the end effector are available. The simulation was carried out after obtaining real image data from points on the end-effector. A CCD camera with focal length .0125 has been used to acquire images. The pixels of selected points are transferred to a host computer for data processing, where multisensor fusion algorithm has been implemented. We choose a specific configuration of the end effector so that the heights of the two feature points on the end effector is the same. This is done in order to obtain an unique solution to the proposed two point algorithm. The real data from one image is observed as follows.

Reference point:	(114.00, 52.00)
Disc center:	(123.00, 116.00)
Two points on the part:	(128.00, 78.00) (108.00, 89.00)
Two points on the end-effector:	(39.00, 200.00) (39.00, 177.00).

The real data is transformed after the virtual rotation of the camera to a set of new coordinates in the transformed camera-frame given as follows.

Reference point:	(0.017947, -0.002470)
Disc center:	(0.016887, 0.000000)
Two points on the part:	(0.016912, -0.0014297) (0.017980, -0.000978)
Two points on the end-effector:	(0.021648, 0.004136) (0.021911, 0.003148)

where the numbers refer to distances in meters.

Note that the Disc Center is the Center of the turntable and the Reference Point is an arbitrary point on the turntable. The axis joining the disc center with the reference point defines an axis with respect to which the rotation of the turntable is measured. The coordinates on the feature points on the endeffector with respect to the transformed camera-frame is transformed to the fixed disc-frame utilizing information from the disc encoder. Finally the coordinates of the feature points on the endeffector with respect to the base frame of the robot is obtained utilizing the robot encoders. Finally the transformation between the robot-frame and the disc-frame is obtained. In the base frame, the co-ordinates of the disc center, known as the origin of the fixed disc frame is (0.6729, -0.3370, -0.4300) and the rotation matrix is given by

$$R = \begin{bmatrix} -0.1888 & 0.9820 & 0 \\ -0.9820 & -0.1888 & 0 \\ 0 & 0 & 1 \end{bmatrix}.$$

The results match very close to the actual values. For the case illustrated in section 2.3, we assume that three points on the end effector are available. We test our scheme for synthetic data below. Suppose that the three points in 3D space, say P_1, P_2 and P_3 satisfy $\overline{P_1P_2} = 0.2070$, $\overline{P_1P_3} = 0.1392$, $\overline{P_2P_3} = 0.1393$ and the projections of $\overline{P_1P_2}$, $\overline{P_1P_3}$ and $\overline{P_2P_3}$ on the horizontal plane are 0.1250, 0.0625 and 0.0625, respectively. Assume that the individual correspondance of the real points and their images, labelled by e_1, e_2 and e_3, is unknown and that in the transformed image plane (after virtually rotating the camera via the algorithm mentioned above), the coordinates of e_i $(i = 1, 2, 3)$ are (0.0216,0.0041), (0.0222,0.0042) and (0.0221,0.0033). Because the points are in front of the camera, there are only three possibilities of the individual correspondance between the points and their images, i.e. $(P_1, P_2, P_3) \rightarrow (e_1, e_2, e_3)$, $(P_1, P_2, P_3) \rightarrow (e_2, e_3, e_1)$ and $(P_1, P_2, P_3) \rightarrow (e_3, e_1, e_2)$, respectively. As stated in the previous sections, the solution for the z-coordinates of each of the three points in the camera frame is unique and therefore the distance of each point on the end-effector from the disc can be uniquely determined since after virtually rotating the camera one can easily obtain the distance of the optical center from the turntable. Let z be the smallest z-coordinate of the three points. By means of the scheme given in the paper, one can obtain three qudratic equations for one unknown z, say $f_1(z) = 0$, $f_2(z) = 0$ and $f_3(z) = 0$, for each of individual correspondance. The solution of the z-coordinate must be a common root of f_1, f_2 and f_3. The simulation result has been sketched in Fig.3–Fig.5, where the values of f_1, f_2 and f_3 are shown as functions of z. It is easily seen that only the case where the individual correspondance is $(e_1, e_2, e_3) \rightarrow (P_1, P_2, P_3)$ is possible and the solution to our problem is $z = 1.225$.

7 Conclusion

To conclude, in this paper we introduce a new calibration free approach to *visual sensing, event based planning* and *tracking control* for the eventual purpose of grasping a part on a moving turntable. The scheme is robust with respect to specific location of the camera and the robot manipulator and the execution of the algorithm is recursive. In this paper we have simulated most of the proposed scheme with real data, i.e. from data as observed by the CCD camera. Experimental implementation of the controller in the closed loop is a topic of our immediate future research.

References

[1] L. Weiss, A. Sanderson and C. Neuman, Dynamic sensor based control of robots with visual feedback, IEEE Trans. Robotics and Automation, RA-3(5): 404-417, October 1987.

[2] J. T. Feddema, C. S. G. Lee and O. R. Mitchell, "Feature-Based Visual Servoing of Robotic Systems," Visual Servoing: Real Time Control of Robot Manipulators Based on Visual Sensory Feedback, Ed. K. Hashimoto, World Scientific Publishing Co. Pt. Ltd, Singapore, 1993.

[3] P. I. Corke, Visual Control of robot manipulators - a review, Visual Servoing, pp. 1 - 32, K. Hashimoto, editor, World Scientific, 1994.

[4] N. Papanikolopoulos, P. K. Khosla and T. Kanade, Visual Tracking of a moving target by a camera mounted on a robot: a combination of control and vision, IEEE Trans. on Robotics and Automation, 9(1), pp. 14-35, 1993.

[5] K. Hashimoto, T. Kimoto, T. Ebine and H. Kimura, Manipulator control with image-based visual servo, Proc. IEEE International Conference on Robotics and Automation, pp. 2267-2271, 1991.

[6] Zhenyu Yu, B. K. Ghosh, N. Xi and T. J. Tarn, "Temporal and Spatial Sensor Fusion in a Robotic Manufacturing Workcell," Proc. of the 1995 IEEE International Conf. on Robotics and Automation, Nagoya, Japan, May 21, 1995.

[7] Zhenyu Yu, B. K. Ghosh, N. Xi and T. J. Tarn, "Multi-Sensor Based Planning and Control for Robotics Manufacturing Systems," Proc. of the 1995 IEEE/RSJ International Conf. on Intelligent Robotics and Systems, Aug. 5–9, 1995.

[8] Zhenyu Yu, B. K. Ghosh, N. Xi and T. J. Tarn, "Visually Controlled Manipulation of Parts in a Robotic Manufacturing Workcell," submitted to IEEE Trans. on Robotics and Automation.

[9] T. J. Tarn, A. K. Bejczy and N. Xi, "Motion Planning in Phase Space for Intelligent Robot Arm Control," Proc. of IROS '92, Raleigh, N.C., 1992, Vol. 3, pp. 1507–1514.

[10] B. K. Ghosh, T. J. Tarn, N. Xi, Z. Yu and D. Xiao, "Calibration free visually controlled manipulation of parts in a robotic manufacturing workcell," Proc. of the 1996 IEEE International Conf. on Robotics and Automation, Minnesota, U.S.A.

[11] K. Kanatani, "Group-Theoretical Methods in Image Understanding," Springer-Verlag, 1990.

[12] T. S. Huang, A. N. Netravali, Motion and structure from feature correspondences: A Review, Proc. of the IEEE, Vol. 82, No. 2, Feb. 1994, pp. 252–268.

[13] R. I. Hartley, Estimation of relative camera positions for uncalibrated cameras, Proc. of 2nd E.C.C.V., Santa Margherita Ligure, Italy, May 1992, pp. 579–587.

[14] M. A. Fischler and R. C. Bolles, Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography, Communications of the ACM, Vol. 24, No. 6, June 1981, pp. 381–355.

[15] B. H. Yoshimi, P. K. Allen, Alignment using an uncalibrated camera system, IEEE Trans. on Robotics and Automation, Vol. 11, No. 4, August 1995, pp. 516–521.

[16] J. Weng, T .S. Huang, N. Ahuja, Motion and structure from image sequences, Springer-Verlag, 1993.

[17] Z. Zhang and O. Faugeras, 3D Dynamic scene analysis, Vol. 27 of Information Sciences, Springer Verlag, 1992.

[18] G. D. Hager and S. Hutchinson, Visual Servoing: Achievements, Issues and Applications, Workshop on Visual Servoing: Achievement, Applications and Open Problems, 1994 IEEE International Conference on Robotics and Automation.

[19] S. Maybank, Theory of Reconstruction from Image Motion, Springer Verlag, 1993.

[20] Y. Wu, S. S. Iyengar, R. Jain, S. Bose, A new generalized computational framework for finding object orientation using perspective trihedral angle constraint, IEEE Trans. on PAMI, Vol. 16, No. 10, Oct. 1994, pp. 961–975.

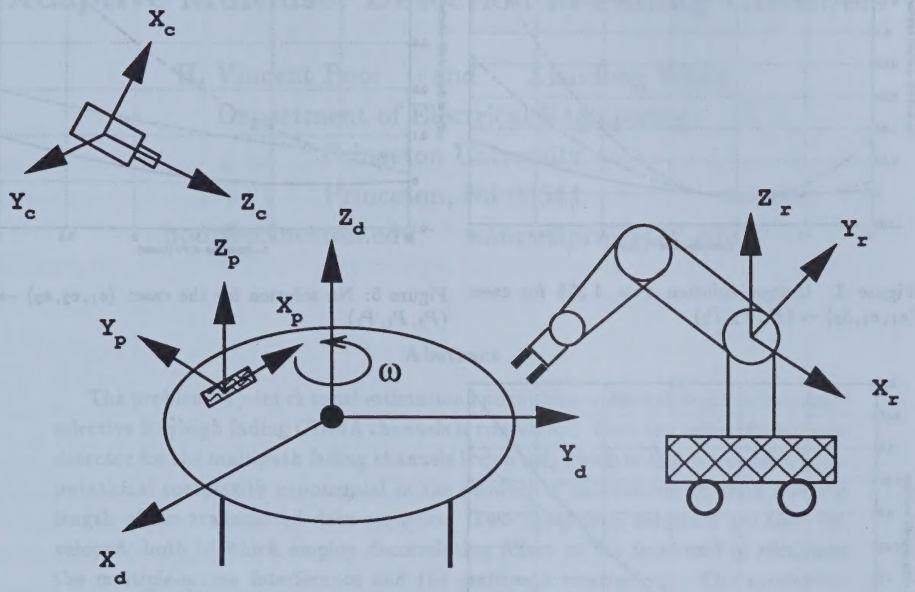


Figure 1: Visually Guided Robotic Manipulation in a Dynamic Environment.

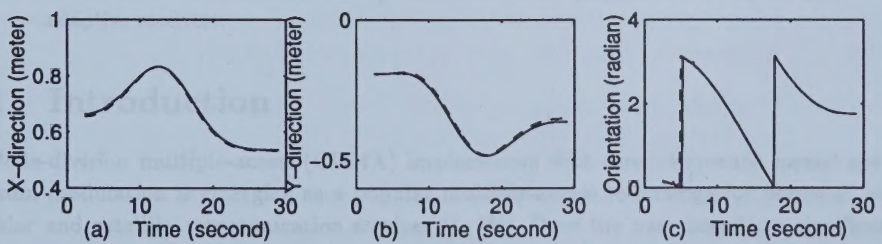


Figure 2: Experimental results

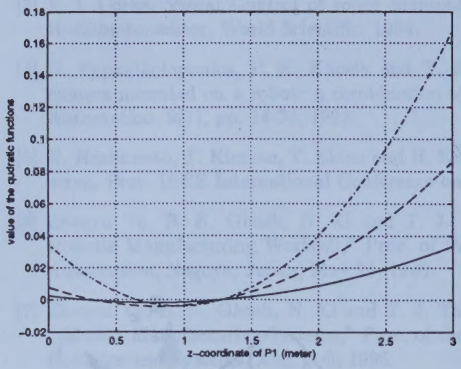


Figure 3: Unique solution $z = 1.225$ for case: $(e_1, e_2, 3_3) \rightarrow (P_1, P_2, P_3)$

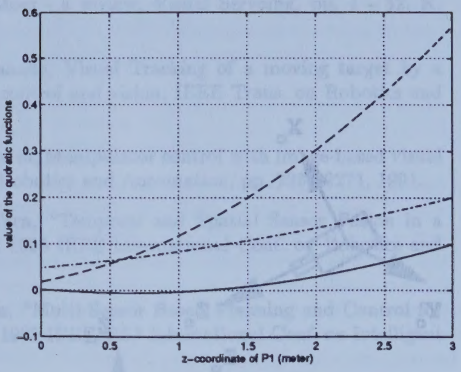


Figure 5: No solution for the case: $(e_1, e_2, e_3) \rightarrow (P_3, P_1, P_2)$

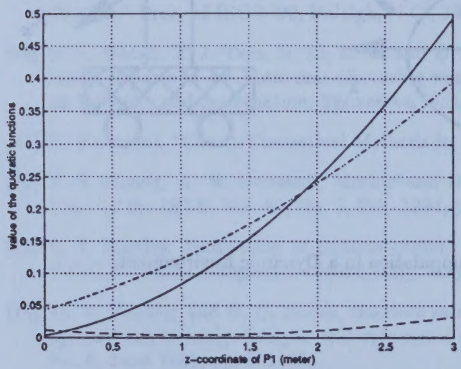


Figure 4: No solution for the case: $(e_1, e_2, e_3) \rightarrow (P_2, P_3, P_1)$

Adaptive Multiuser Detection in Fading Channels *

H. Vincent Poor and Xiaodong Wang
Department of Electrical Engineering
Princeton University
Princeton, NJ 08544
poor@princeton.edu wangx@princeton.edu

Abstract

The problem of joint channel estimation and multiuser detection in the frequency-selective Rayleigh fading CDMA channels is considered. First the *optimal* multiuser detector for the multipath fading channels is derived, which is shown to have a computational complexity exponential in the product of the number of users and the length of the transmitted data sequence. Two *suboptimal* detectors are then developed, both of which employ decorrelating filters at the front-end to eliminate the multiple-access interference and the multipath interference. The *symbol-by-symbol* detector uses a Kalman filter and decision feedback to track the fading channel for diversity combing. The *per-survivor sequence* detector is in the form of a Viterbi algorithm with the trellis updates being computed by a bank of Kalman filters in the per-survivor fashion. *Adaptive* versions of these suboptimal detectors that require the knowledge of only the waveform of the user of interest are also developed. The adaptive receivers employ recursive-least-squares (RLS) minimum-mean-square-error (MMSE) filters at the front-end to mitigate the interference, and use a bank of linear predictors to track the fading channels. Simulation results are provided to demonstrate the performance of the proposed non-adaptive and adaptive receivers.

1 Introduction

Code-division multiple-access (CDMA) implemented with direct-sequence spread spectrum modulation is emerging as a popular multiple-access technology for personal, cellular and satellite communication services [7, 15]. Over the past decade, a significant amount of research has addressed the problems of optimal as well as near-optimal detection of CDMA signals over the additive white Gaussian noise (AWGN) channel (see [13] for a review). More recently, adaptive multiuser detection has been addressed for the AWGN channel (see [9, 14] for reviews). In many CDMA communications applications, such as cellular mobile radio systems, indoor wireless communications and underwater

*This research was supported by the U. S. Army Research Office under Grant DAAH04-93-G-0219.

local area networks, the channels exhibit severe multipath fading, which limits the system performance. Recent works on multiuser detection have addressed the multiple-access fading channels [2, 6, 12, 17]. In this paper, we develop *optimal* and *suboptimal* as well as *adaptive* multiuser detectors for joint channel estimation and data detection in the frequency-selective Rayleigh fading CDMA channels.

2 System Descriptions

Consider a K-user binary communication system, employing normalized modulation waveforms $s_1, s_2, \cdots, s_K$, and signaling through a frequency-selective Rayleigh fading channel. It is assumed that the signaling interval T is smaller than the channel coherence time, such that the fading attenuation can be assumed to be constant for the duration of at least one signaling interval. Under this slow fading condition, the received signal can be modeled as

$$r(t) = \sum_{i=0}^{M-1} \sum_{k=1}^{K} e^{j\phi_k} A_k b_k(i) \sum_{l=1}^{L} c_{kl}(i) s_k \left(t - (i-1)T - \tau_k - \frac{l-1}{W} \right) + n(t), \tag{1}$$

where $n(t)$ is a zero mean complex white Gaussian noise process with independent real and imaginary components, each with power spectral density $\sigma^2/2$; M is the number of data bits transmitted; A_k, ϕ_k and τ_k are the amplitude, carrier phase and relative delay, respectively, of the k-th user signal; $b_k(i) \in \{+1, -1\}$ is the i-th bit of the k-th user; T is the signaling interval; W is the bandwidth of the spread-spectrum signal; L is the number of resolvable paths for each user; and $c_{kl}(i)$ is the complex channel gain process for the l-th path of the k-th user signal. The KL channel fading processes $\{c_{kl}(i), 1 \le k \le K, 1 \le l \le L\}$ are assumed to be statistically independent of each other. It is assumed that s_k, the k-th user's normalized signaling waveform, is supported only on the interval $[0, T]$, and that $\{b_k(i)\}$ is a collection of independent equiprobable binary (± 1) random variables.

The Rayleigh fading coefficients $\{c_{kl}(i)\}$ are modeled as zero-mean complex Gaussian random variables. The fading coefficients $\{c_{kl}(i)\}$, which approximate the statistical properties of Rayleigh fading model, can be generated by feeding white complex Gaussian noise $\{e_{kl}(i)\}$ through a linear filter, given in a state space form by,

$$\underline{x}_{kl}(i+1) = \underline{F}\underline{x}_{kl}(i) + \underline{g}e_{kl}(i), \tag{2}$$

$$c_{kl}(i) = \underline{h}\underline{x}_{kl}(i), \qquad k = 1, \cdots, K; \quad l = 1, \cdots, L. \tag{3}$$

The sequences $\{e_{kl}(i)\}$ are mutually independent white complex Gaussian noise, with zero means and variances q_{kl}. The matrix $\underline{F}$ and vectors $\underline{g}$ and $\underline{h}$ are real valued.

3 Optimal Detectors

The result in this section is a generalization of the optimal receiver for the single-user single-path fading channel developed in [3] to the multiuser multipath fading channels. We start by considering the synchronous case of model (1), in which $\tau_1 = \tau_2 = \cdots =$

$\tau_K = 0$. For the data signaling interval much longer than the multipath delay spread, i.e., $T \gg T_m \approx \frac{L}{W}$, any intersymbol interference (ISI) due to channel dispersion can be neglected [10]. Therefore, the synchronous case of model (1) can be viewed as equivalent to a synchronous multiuser system with KL users,

$$r(t) = \sum_{i=0}^{M-1} \sum_{k=1}^{K} e^{j\phi_k} A_k b_k(i) \sum_{l=1}^{L} c_{kl}(i) s_{kl}(t - iT) + n(t), \tag{4}$$

where $s_{kl}(t) \triangleq s_k(t - \frac{l-1}{W})$. At the receiver, a bank of matched filters is formed and the output of the matched-filter-bank is given by

$$\mathbf{y}(i) = \mathbf{R}\mathbf{\Phi}\mathbf{A}\mathbf{B}(i)\mathbf{c}(i) + \mathbf{n}(i), \tag{5}$$

where $[\mathbf{y}(i)]_{(k-1)L+l} \triangleq \int_{iT}^{(i+1)T} s_{kl}(t - iT) r(t) dt$; $\mathbf{R}$ is the $KL \times KL$ normalized cross-correlation matrix, $[\mathbf{R}]_{(m-1)L+n,(k-1)L+l} = \int_0^T s_{mn}(t) s_{kl}(t) dt$; $\mathbf{\Phi} \triangleq \mathrm{diag}(e^{j\phi_1}\mathbf{I}, \cdots, e^{j\phi_K}\mathbf{I})$, $\mathbf{A} \triangleq \mathrm{diag}(A_1\mathbf{I}, \cdots, A_K\mathbf{I})$, $\mathbf{B}(i) \triangleq \mathrm{diag}(b_1(i)\mathbf{I}, \cdots, b_K(i)\mathbf{I})$, $\mathbf{I}$ is the $L \times L$ identity matrix; $\mathbf{c}(i)$ is the KL-vector of channel gain, $[\mathbf{c}(i)]_{(k-1)L+l} = c_{kl}(i)$; and $\mathbf{n}(i)$ is a complex Gaussian random vector, with independent $\mathcal{N}(\mathbf{0}, \frac{\sigma^2}{2}\mathbf{R})$ real and imaginary components. Note that since the channel introduces memory, the whole received signal needs to be utilized in detecting each data bit even when all users' signals are synchronous.

Denote $\mathbf{x}(i) \triangleq \left[\underline{x}_{11}^T(i) \cdots \underline{x}_{1L}^T(i) \cdots \cdots \underline{x}_{K1}^T(i) \cdots \underline{x}_{KL}^T(i)\right]^T$, $\mathbf{e}(i) \triangleq [e_{11}(i) \cdots e_{1L}(i) \cdots \cdots e_{K1}(i) \cdots e_{KL}(i)]^T$, $\mathbf{F} \triangleq \mathrm{diag}(\underbrace{\underline{F}, \cdots, \underline{F}}_{KL})$, $\mathbf{G} \triangleq \mathrm{diag}(\underbrace{\underline{g}, \cdots, \underline{g}}_{KL})$, $\mathbf{H} \triangleq \mathrm{diag}(\underbrace{\underline{h}, \cdots, \underline{h}}_{KL})$.

Then the channel fading model (2) and (3) can be expressed in the vector form as

$$\mathbf{x}(i+1) = \mathbf{F}\mathbf{x}(i) + \mathbf{G}\mathbf{e}(i), \tag{6}$$

$$\mathbf{c}(i) = \mathbf{H}\mathbf{x}(i), \tag{7}$$

where the input sequence $\{\mathbf{e}(i)\}$ is a sequence of independent complex Gaussian random vectors, each $\mathbf{e}(i) \sim \mathcal{N}_c(\mathbf{0}, \mathbf{Q})$, where $\mathbf{Q} \triangleq \mathrm{diag}(q_{11} \cdots q_{1L} \cdots \cdots q_{K1} \cdots q_{KL})$. Substituting (7) into (5), we get

$$\mathbf{y}(i) = \mathbf{R}\mathbf{\Phi}\mathbf{A}\mathbf{B}(i)\mathbf{H}\mathbf{x}(i) + \mathbf{n}(i). \tag{8}$$

Denote $\mathbf{b}(i) \triangleq [b_1(i), \cdots, b_K(i)]^T$, $\mathcal{B}(i) \triangleq \{\mathbf{b}(1), \mathbf{b}(2), \cdots, \mathbf{b}(i)\} = \mathcal{B}(i-1) \cup \{\mathbf{b}(i)\}$, and $\mathcal{Y}(i) \triangleq \{\mathbf{y}(1), \mathbf{y}(2), \cdots, \mathbf{y}(i)\} = \mathcal{Y}(i-1) \cup \{\mathbf{y}(i)\}$, $i = 1, \cdots, M$. Then the MAP estimate $\hat{\mathcal{B}}(M)$ selects the sequence $\hat{\mathcal{B}}(M)$ whose *a posteriori* probability is maximum:

$$\hat{\mathcal{B}}(M) = \arg \max_{\mathcal{B}^m(M) \in \{-1,1\}^{K \times M}} P\left[\mathcal{B}^m(M) \mid \mathcal{Y}(M)\right]. \tag{9}$$

Let $\mathcal{B}^m(M)$ be any one possible transmitted sequence of data bit vectors. It is straightforward to show that

$$P[\mathcal{B}^m(i) \mid \mathcal{Y}(i)] = \frac{P[\mathcal{B}^m(i-1) \mid \mathcal{Y}(i-1)]\, p[\mathbf{y}(i) \mid \mathcal{Y}(i-1), \mathcal{B}^m(i)]}{2^K\, p[\mathbf{y}(i) \mid \mathcal{Y}(i-1)]}. \tag{10}$$

By (6) and (8), the second term in the numerator of (10) is a complex Gaussian pdf, with mean $\hat{\mathbf{y}}_{i|i-1}^m \triangleq E\{\mathbf{y}(i) \mid \mathcal{Y}(i-1), \mathcal{B}^m(i)\}$, and covariance matrix $\hat{\Sigma}_{i|i-1}^m \triangleq E\left\{\left[\mathbf{y}(i) - \hat{\mathbf{y}}_{i|i-1}^m\right]\right.$

$\left[\mathbf{y}(i) - \hat{\mathbf{y}}^m_{i|i-1}\right]^H \mid \mathcal{Y}(i-1), \mathcal{B}^m(i)\Big\}$, which can be calculated recursively by the Kalman filter [1]. The denominator of (10) is independent of $\mathcal{B}^m(i)$ and thus has no effect on the maximization in (9). With (10) a recursive procedure for calculating the *a posteriori* probability (10) has been specified, and hence the optimal receiver for joint multiuser detection and channel estimation is obtained. For the maximization in (9), each possible sequence of data vectors must be tested, and a different Kalman filter is needed for each assumed data vector sequence. The computational complexity of the optimal receiver is therefore $O\left(2^{MK}\right)$.

It can be shown that the optimal receiver structure for the asynchronous channel is the same as that for the synchronous channel (thus has complexity $O\left(2^{MK}\right)$), except that in that case the Kalman filtering is carried out in the *colored* input noise and *colored* observation noise.

4 Suboptimal Detectors

4.1 Symbol-by-symbol Detector

The suboptimal receiver considered here employs a decorrelating filter at the front-end to eliminate the multiple-access interference and the multipath interference. Suppose the user of interest is user 1. Let $\mathbf{z}_1(i)$ be the subvector of the decorrelating filter output corresponding to the L-path signals of the first user, then

$$\mathbf{z}_1(i) = e^{j\phi_1} A_1 b_1(i) \mathbf{c}_1(i) + \mathbf{n}_1(i) \tag{11}$$

where $\mathbf{c}_1 \triangleq [c_{11}(i), \cdots, c_{1L}(i)]^T$, $\mathbf{n}_1(i)$ is a complex Gaussian vector with independent $\mathcal{N}(\mathbf{0}, \frac{\sigma^2}{2}\mathbf{S})$ real and imaginary components, with $\mathbf{S} = [\mathbf{R}^{-1}]_{1:L,\,1:L}$.

The channel gain sequence $\{\mathbf{c}_1(i)\}$ satisfies the linear Gaussian model

$$\mathbf{x}_1(i+1) = \mathbf{F}_1 \mathbf{x}_1(i) + \mathbf{G}_1 \mathbf{e}_1(i), \tag{12}$$

$$\mathbf{c}_1(i) = \mathbf{H}_1 \mathbf{x}_1(i), \tag{13}$$

where $\mathbf{x}_1(i) \triangleq \left[\underline{x}^T_{11}(i) \cdots \underline{x}^T_{1L}(i)\right]^T$, $\mathbf{e}_1(i) \triangleq [e_{11}(i) \cdots e_{1L}(i)]^T$, $\mathbf{F}_1 \triangleq \mathrm{diag}(\underbrace{\underline{F}, \cdots, \underline{F}}_{L})$, $\mathbf{G}_1 \triangleq \mathrm{diag}(\underbrace{\underline{g}, \cdots, \underline{g}}_{L})$, $\mathbf{H}_1 \triangleq \mathrm{diag}(\underbrace{\underline{h}, \cdots, \underline{h}}_{L})$. Assume that at time i, all the previously transmitted symbols $b_1(1), \cdots, b_1(i-1)$ are known. Substituting (13) into (11) we get

$$\mathbf{z}_1(i) = e^{j\phi_1} A_1 b_1(i) \mathbf{H}_1 \mathbf{x}_1(i) + \mathbf{n}_1(i). \tag{14}$$

The equivalent expressions for (12) and (14) in the real domain are given by (notice that $\mathbf{F}_1$, $\mathbf{G}_1$ and $\mathbf{H}_1$ are all real quantities.)

$$\underbrace{\begin{bmatrix} \mathbf{x}^R_1(i+1) \\ \mathbf{x}^I_1(i+1) \end{bmatrix}}_{\tilde{\mathbf{x}}(i+1)} = \underbrace{\begin{bmatrix} \mathbf{F}_1 & \\ & \mathbf{F}_1 \end{bmatrix}}_{\tilde{\mathbf{F}}} \underbrace{\begin{bmatrix} \mathbf{x}^R_1(i) \\ \mathbf{x}^I_1(i) \end{bmatrix}}_{\tilde{\mathbf{x}}(i)} + \underbrace{\begin{bmatrix} \mathbf{G}_1 & \\ & \mathbf{G}_1 \end{bmatrix}}_{\tilde{\mathbf{G}}} \underbrace{\begin{bmatrix} \mathbf{e}^R_1(i) \\ \mathbf{e}^I_1(i) \end{bmatrix}}_{\tilde{\mathbf{e}}(i)}, \tag{15}$$

$$\underbrace{\begin{bmatrix} \mathbf{z}_1'^{R}(i) \\ \mathbf{z}_1'^{I}(i) \end{bmatrix}}_{\bar{\mathbf{z}}(i)} = A_1 b_1(i) \underbrace{\begin{bmatrix} \mathbf{H}_1 & \\ & \mathbf{H}_1 \end{bmatrix}}_{\bar{\mathbf{H}}} \underbrace{\begin{bmatrix} \mathbf{x}_1^R(i) \\ \mathbf{x}_1^I(i) \end{bmatrix}}_{\bar{\mathbf{x}}(i)} + \begin{bmatrix} \mathbf{n}_1'^{R}(i) \\ \mathbf{n}_1'^{I}(i) \end{bmatrix}, \tag{16}$$

where the superscripts ^{R} and ^{I} represent the real and imaginary components of a complex vector; $\mathbf{z}_1' \triangleq \mathbf{z}_1 e^{-j\phi_1}$; $\mathbf{n}_1' \triangleq \mathbf{n}_1 e^{-j\phi_1}$; $\bar{\mathbf{e}}(i) \sim \mathcal{N}\left(\mathbf{0}, \bar{\mathbf{Q}}\right)$, with $\bar{\mathbf{Q}} \triangleq \text{diag}(\frac{1}{2}q_{11}, \cdots, \frac{1}{2}q_{1L}, \frac{1}{2}q_{11}, \cdots, \frac{1}{2}q_{1L})$. Given $\{b_1(1), \cdots, b_1(i-1)\}$ and $\{\mathbf{z}_1(1), \cdots, \mathbf{z}_1(i-1)\}$, $\bar{\mathbf{x}}(i)$ is a Gaussian vector with mean $\hat{\bar{\mathbf{x}}}_{i|i-1}$ and covariance matrix $\hat{\bar{\mathbf{\Sigma}}}_{i|i-1}$, which can be calculated recursively by the Kalman filter as follows

$$\mathbf{K}_i = A_1 \hat{\bar{\mathbf{\Sigma}}}_{i|i-1} \bar{\mathbf{H}}^T \left(A_1^2 \bar{\mathbf{H}} \hat{\bar{\mathbf{\Sigma}}}_{i|i-1} \bar{\mathbf{H}}^T + \bar{\mathbf{Q}}\right)^{-1}, \tag{17}$$

$$\hat{\bar{\mathbf{x}}}_{i+1|i} = \bar{\mathbf{F}} \hat{\bar{\mathbf{x}}}_{i|i-1} + \bar{\mathbf{F}} \mathbf{K}_i \left(b_1(i) \bar{\mathbf{z}}(i) - A_1 \bar{\mathbf{H}} \hat{\bar{\mathbf{x}}}_{i|i-1}\right), \tag{18}$$

$$\hat{\bar{\mathbf{\Sigma}}}_{i+1|i} = \bar{\mathbf{F}} \left(\mathbf{I} - A_1 \mathbf{K}_i \bar{\mathbf{H}}\right) \hat{\bar{\mathbf{\Sigma}}}_{i|i-1} \bar{\mathbf{F}}^T + \bar{\mathbf{G}} \bar{\mathbf{Q}} \bar{\mathbf{G}}^T. \tag{19}$$

The maximum likelihood detection of $b_1(i)$ is then given by

$$\hat{b}_1(i) = \text{sgn}\left[\hat{\bar{\mathbf{x}}}_{i|i-1}^T \bar{\mathbf{H}}^T \left(A_1^2 \bar{\mathbf{H}} \hat{\bar{\mathbf{\Sigma}}}_{i|i-1} \bar{\mathbf{H}}^T + \sigma^2 \bar{\mathbf{S}}\right)^{-1} \bar{\mathbf{z}}(i)\right]. \tag{20}$$

Note that when the channel is perfectly known, i.e., $\hat{\bar{\mathbf{x}}}_{i|i-1} = \bar{\mathbf{x}}(i)$ and $\hat{\bar{\mathbf{\Sigma}}}_{i|i-1} = \mathbf{0}$, then it can be verified that (20) reduces to

$$\hat{b}_1(i) = \text{sgn}\left(\text{Re}\left[e^{-j\phi_1} \mathbf{c}_1^H(i) \mathbf{S}^{-1} \mathbf{z}_1(i)\right]\right), \tag{21}$$

a detector considered in [5, 16].

4.2 Per-survivor Sequence Detector

In this subsection, we develop an approximate MAP sequence detector for detecting the symbol sequence $\{b_1(i)\}$ from the decorrelator output sequence $\{\mathbf{z}_1(i)\}$. The proposed per-survivor sequence detector [11] proceeds as follows. A trellis structure is defined first. The trellis has 2^Q states and each state Δ_i represents the reduced channel state, corresponding to one of the possible vectors of the current and the $Q-1$ previous data bits at the i-th signaling interval, so that

$$\Delta_i = [\hat{\bar{\mathbf{x}}}_{i+1|i}, \hat{\bar{\mathbf{\Sigma}}}_{i+1|i};\ \hat{b}_1(i), \hat{b}_1(i-1), \cdots, \hat{b}_1(i-Q+1)], \tag{22}$$

where $\hat{\bar{\mathbf{x}}}_{i+1|i}$ and $\hat{\bar{\mathbf{\Sigma}}}_{i+1|i}$ are the values of the Kalman predictor based on the data bit sequence associated with the *survivor* path entering the state Δ_i. For each state Δ_i there are 2 transitions emerging from it and going to 2 different states Δ_{i+1}: each transition corresponds to one of the two possible choices for the symbol $\hat{b}_1(i+1)$. Let the transition be denoted by $\Delta_i \longrightarrow \Delta_{i+1}$. A branch metric $\lambda[\Delta_i \longrightarrow \Delta_{i+1}]$ is associated with each transition and is defined as

$$\begin{aligned} \lambda[\Delta_i \longrightarrow \Delta_{i+1}] = & \left[\bar{\mathbf{z}}(i+1) - A_1 \hat{b}_1(i+1) \bar{\mathbf{H}} \hat{\bar{\mathbf{x}}}_{i+1|i}\right]^T \left(A_1^2 \bar{\mathbf{H}} \hat{\bar{\mathbf{\Sigma}}}_{i+1|i} \bar{\mathbf{H}}^T + \sigma^2 \bar{\mathbf{S}}\right)^{-1} \\ & \cdot \left[\bar{\mathbf{z}}(i+1) - A_1 \hat{b}_1(i+1) \bar{\mathbf{H}} \hat{\bar{\mathbf{x}}}_{i+1|i}\right] + \log \left|A_1^2 \bar{\mathbf{H}} \hat{\bar{\mathbf{\Sigma}}}_{i+1|i} \bar{\mathbf{H}}^T + \sigma^2 \bar{\mathbf{S}}\right| \end{aligned} \tag{23}$$

A one-step Viterbi algorithm is then performed to calculate the accumulated metric at the state Δ_{i+1}, according to

$$\Lambda[\Delta_{i+1}] = \min_{\Delta_i} \{\Lambda[\Delta_i] + \lambda[\Delta_i \longrightarrow \Delta_{i+1}]\}. \tag{24}$$

The channel estimates $\hat{\mathbf{x}}_{i+2|i+1}$ and $\hat{\Sigma}_{i+2|i+1}$ at the state Δ_{i+1} are then updated using the Kalman filter (17)–(19) based on the estimate at time i from the state Δ_i that complies with (24), with the result that they are associated with the *survivors* as opposed to the *states*.

5 Adaptive Detectors

The suboptimal detectors developed in the previous section require the knowledge of all waveforms of all users in the channel in order to implement the front-end decorrelating filters, and the channel fading parameters in order to implement the Kalman filters. This information requirement is certainly too stringent for practical applications. Next we develop their adaptive versions which require knowledge only of the waveform of the user of interest.

5.1 Front-end MMSE Filter

The adaptive detectors employ a recursive least-squares (RLS) version of the multipath multiuser MMSE filter as the front-end to mitigate the multiple-access interference and the multipath interference. Such an adaptive front-end filter is termed *blind* as its implementation requires the knowledge of the waveform of only the user of interest [4]. Henceforth we assume a vector representation for the signals, which results from sampling the received signals at the chip-rate. Let $\mathbf{r}(i)$ be the complex vector of the chip-rate sampled received signal during the i-th signaling interval, and $\mathbf{s}_1, \cdots, \mathbf{s}_L$ be the vector representation of the L-path real signature waveforms for the user of interest, where L is the maximum number of resolvable paths. Let $\mathbf{w}_l(i)$ be the tap weight vector at time i of the MMSE filter for the l-th path. The exponentially windowed RLS algorithm selects the weight vectors $\mathbf{W}(n) \triangleq [\mathbf{w}_1(n) \cdots \mathbf{w}_L(n)]$ which

$$\begin{aligned} \text{minimize} \quad & \sum_{i=1}^{n} \lambda^{n-i} \left\| \mathbf{W}^H(n)\mathbf{r}(i) \right\|^2, \\ \text{subject to} \quad & \mathbf{s}_l^T \mathrm{Re}\{\mathbf{w}_l(n)\} = 1, \quad \mathbf{s}_l^T \mathrm{Im}\{\mathbf{w}_l(n)\} = 1, \quad 1 \le l \le L, \end{aligned}$$

where $0 < \lambda < 1$ is the forgetting factor ($1 - \lambda \ll 1$). Let the output of the RLS multipath MMSE filter be $\mathbf{z}(n) \triangleq \mathbf{W}^H(n)\mathbf{r}(n) \triangleq [z_1(n) \cdots z_L(n)]^T$. Then

$$z_l(n) = Ab(n)c_l(n) + v_l(n), \quad 1 \le l \le L, \tag{25}$$

where A and $b(n)$ are respectively the signal amplitude and the data bit at time n of the user of interest; $c_l(n)$ is the l-th channel fading gain; $v_l(n)$ is a noise term which is *approximately Gaussian* [8]. A QR-based RLS algorithm for computing $\mathbf{z}(n)$ can be

derived, which leads to a systolic array implementation of the front-end RLS multipath MMSE filter that exploits massively parallel signal processing computation. The details of this QR-RLS algorithm will be reported elsewhere.

5.2 Symbol-by-symbol Detection

In the absence of channel statistics, we adopt the simple combining strategy for symbol detection:

$$\hat{b}(i) = \text{sgn}\left(\text{Re}\left\{\hat{\mathbf{c}}^H(i)\mathbf{z}(i)\right\}\right), \tag{26}$$

where $\hat{\mathbf{c}}(i)$ is the estimated channel fading gain vector at time i. A linear predictor of order P with decision-feedback is employed for tracking each fading channel:

$$\hat{c}_l(i) = \sum_{j=1}^{P} a_{lj}\hat{b}(i-j)z_l(i-j). \tag{27}$$

The coefficients $\{a_{lj}\}$ of the linear predictors are computed adaptively using the LMS algorithm.

5.3 Per-survivor Sequence Detection

The adaptive per-survivor sequence detector based on the output of the front-end MMSE filter has the same structure as discussed in the previous section, except that the bank of Kalman filters is replaced by a bank of linear predictors. The channel memory is truncated, and the effects of the residual channel memory on the calculation of each transition metric in the Viterbi algorithm is replaced by that of the data sequence associated with the survivor leading to that transition. The linear predictors are used to track the fading channels, each associating with one survivor.

6 Simulation Results

A synchronous three-user ($K = 3$) system with processing gain $N = 31$ is employed in the simulations. The number of resolvable paths is $L = 3$. The Rayleigh fading process $\{c_{kl}(i)\}$ is generated by filtering two independent real iid Gaussian processes with two identical 3rd-order Butterworth filters. The 3-dB bandwidth f_D of the filter normalized to the symbol rate $1/T$ is used as a measure of the fading rate. In Figure 1 we plot the lower bounds for the error probabilities of the two suboptimal detectors, obtained by providing the Kalman filters with correct data sequence ("known data bound"). To compare, we also plot the "known channel bound" (where the fading gains are assumed to be known), as well as the error probabilities of the conventional RAKE receiver and that of the differentially coherent receiver. In Figure 2 the actual performance of the non-adaptive suboptimal receivers is shown. Note that the use of pilot symbols is mandatory to prevent catastrophic error propagation. In the simulation, every N_d information data bits are preceded by N_p pilot bits. The performance of the adaptive suboptimal receivers is shown in Figure 3.

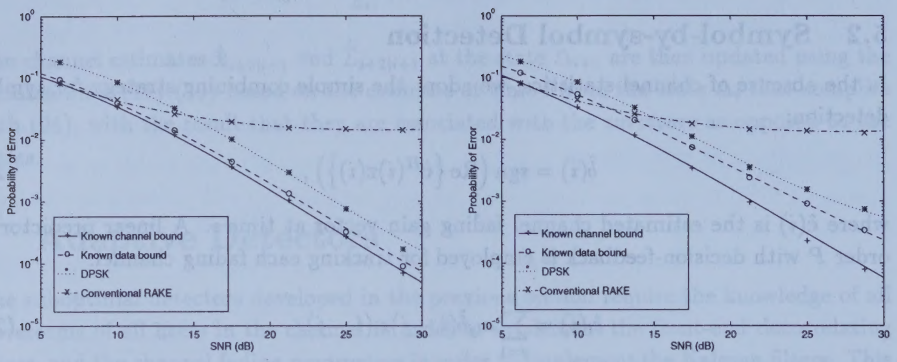


Figure 1: Simulated performance bounds for the non-adaptive suboptimal detectors with $f_D T$ =0.01 (left) and 0.05 (right). The known data bound curve is for both the symbol-by-symbol detector and the per-survivor sequence detector. The number of states in the Trellis is 8 ($Q = 3$) for the per-survivor detector.

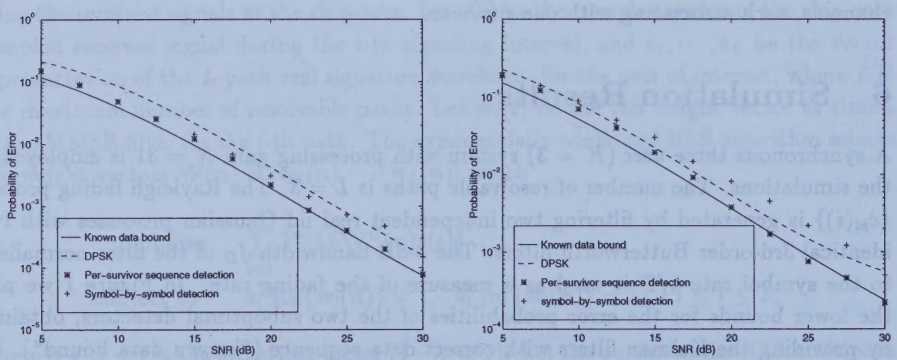


Figure 2: Simulated performance of the non-adaptive suboptimal detectors with $f_D T = 0.01$ (left, $N_d = 50$, $N_p = 10$) and 0.05 (right, $N_d = 10$, $N_p = 10$). The number of states in the trellis is 8 ($Q = 3$) for the per-survivor sequence detector.

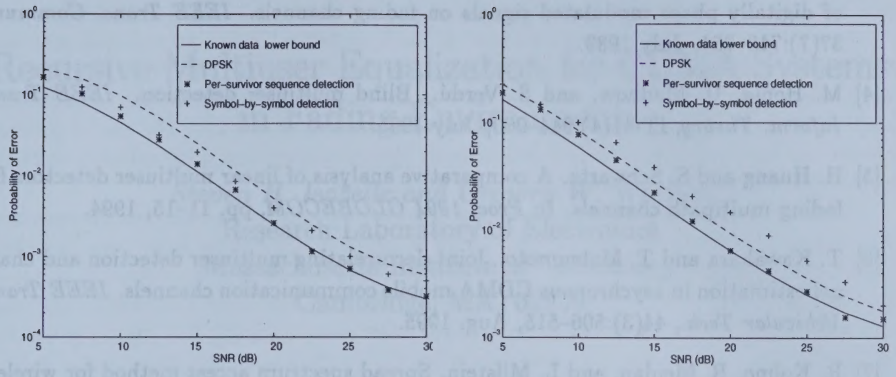


Figure 3: Simulated performance of the adaptive suboptimal detectors with $f_D T = 0.01$ (left, $N_d = 50$, $N_p = 10$) and 0.05 (right, $N_d = 10$, $N_p = 10$). The number of states in the trellis is 8 ($Q = 3$) for the per-survivor sequence detector.

7 Conclusions

In this paper, we have considered the problem of joint multiuser detection and channel estimation for the frequency-selective Rayleigh fading CDMA channels. First we have shown that the *optimal* multiuser detectors for both the synchronous and the asynchronous multipath fading channels have computational complexities exponential in the product of the number of users and the length of the transmitted data sequence. We have then developed two suboptimal detectors, namely, the symbol-by-symbol detector and the per-survivor sequence detector. Unlike previous works, in our treatment of the symbol-by-symbol detector, the effect of the channel estimation error is taken into account. And similar detectors obtained previously [5, 16] correspond to the hypothetical case of our symbol-by-symbol detector when there are no estimation errors. The per-survivor sequence detector is an approximation to the maximum likelihood sequence detector, which performs better than the decision-directed symbol-by-symbol detector, at the expense of higher complexity. Finally, we have developed adaptive versions of the suboptimal detectors that require the signature waveform of only the user of interest. These adaptive receivers employ RLS MMSE filters at the front-end to mitigate interference and use a bank of linear predictors to track the fading channels.

References

[1] B.D.O. Anderson and J.B. Moore. *Optimal Filtering.* Prentice Hall, 1979.

[2] U. Fawer and B. Aazhang. A multiuser receiver for code division multiple access communications over multipath channels. *IEEE Trans. Commun.*, COM-43(2/3/4):1556–1565, Feb/Mar/Apr. 1995.

[3] R. Haeb and H. Meyr. A systematic approach to carrier recovery and detection of digitally phase modulated signals on fading channels. *IEEE Trans. Commun.*, 37(7):748–754, July 1989.

[4] M. Honig, U. Madhow, and S. Verdú. Blind multiuser detection. *IEEE Trans. Inform. Theory*, IT-41(4):944–960, July 1995.

[5] H. Huang and S. Schwartz. A comparative analysis of linear multiuser detectors for fading multipath channels. In *Proc. 1994 GLOBECOM*, pp. 11–15, 1994.

[6] T. Kawahara and T. Matsumoto. Joint decorrelating multiuser detection and channel estimation in asychronous CDMA mobile communication channels. *IEEE Trans. Vehicular Tech.*, 44(3):506–515, Aug. 1995.

[7] R. Kohno, R. Meidan, and L. Milstein. Spread spectrum access method for wireless communications. *IEEE Commun. Mag.*, pp. 58–67, Jan. 1995.

[8] H. V. Poor and S. Verdú. On the bit-error probability of MMSE multiuser detection. To appear in *IEEE Trans. Inform. Theory*, 1997.

[9] H.V. Poor. Adaptivity in multiple-access communications. In *Proc. 34th IEEE Conference on Decision and Control*, New Orleans, LA, Dec. 1995.

[10] J.G. Proakis. *Digital Communications.* McGraw-Hill, 3rd edition, 1995.

[11] R. Raheli, A. Polydoros, and C. Tzou. Per-survivor processing: A general approach to MLSE in uncertain environments. *IEEE Trans. Commun.*, COM-43(2/3/4):354–364, Feb/Mar/Apr 1995.

[12] M.K. Varanasi and S. Vasudevan. Multiuser detectors for synchronous CDMA communication over non-selective Rician fading channels. *IEEE Trans. Commun.*, COM-42(2/3/4):711–722, Feb/Mar/Apr. 1994.

[13] S. Verdú. Multiuser detection. In H. V. Poor and J. B. Thomas, editors, *Advances in Statistical Signal Processing Vol. 2: Signal Detection.* JAI Press, Greenwich, CT, 1993.

[14] S. Verdú. Adaptive multiuser detection. In *Proc. IEEE Int. Symp. on Spread Spectrum Theory and Applications*, Oulu, Finland, July 1994.

[15] A.J. Viterbi. The orthogonal-random wave form dichotomy for digital mobile personal communications. *IEEE Personal Communications Magazine*, 1(1):18–24, Feb. 1994.

[16] Z. Zvonar and D. Brady. Coherent and differentially coherent multiuser detectors for asynchrnous CDMA frequency-selective channels. In *Proc. MILCOM'92*, pp. 442–446, San Diego, CA, Oct. 1992.

[17] Z. Zvonar and D. Brady. Multiuser detection in single-path fading channels. *IEEE Trans. Commun.*, COM-42(2/3/4):1729–1739, Feb/Mar/Apr 1994.

Recursive Multiuser Equalization for CDMA Systems in Fading Environments

Steven H. Isabelle and Gregory W. Wornell
Research Laboratory of Electronics
Massachusetts Institute of Technology
Cambridge, MA 02139

Abstract

A general and efficient framework is developed for addressing problems of multiuser equalization in wireless communication systems operating in the presence of time- and/or frequency-selective fading due to multipath propagation. In particular, a state-space description of the canonical CDMA system as a multiple-input multiple-output linear system is used to obtain minimum mean-square error multiuser equalizers having efficient recursive implementations in the form of Kalman filters.

These algorithms lead to near-far resistant receivers that obviate the need for power control in the system. More generally, the equalizers make efficient use of all available temporal, spectral, and spatial diversity in the system, and can be used in conjunction with both conventional CDMA and more recently proposed spread-signature CDMA systems. Decision-feedback implementations of the equalizers are also described.

1 Introduction

In a wide-range of wireless communication applications, there is a need for users to be able to communicate efficiently and asynchronously among themselves in the presence of fading due to multipath propagation. For example, in a cellular mobile radio environment, the transmissions of the individual mobiles pass through generally distinct channels, and a noisy version of their superposition is obtained at the base station.

In such scenarios, obtaining reliable estimates of the symbols transmitted by a particular user (or all users) requires the mitigation of several sources of interference, including multiple-access interference and intersymbol interference.

From a broader perspective, reliable communication in the presence of fading requires that diversity be exploited to improve both average and worst-case performance (outage probabilities). Such diversity takes three forms: spectral, temporal, and spatial. For example, conventional CDMA systems, where all users spread their transmission over a common bandwidth, is a format designed to enable spectral diversity to be exploited. In particular, provided the total bandwidth is large compared to the coherence bandwidth of the channel, then spectral diversity can be exploited by a suitably designed receiver. In

This work has been supported in part by ARPA monitored by ONR under Contract No. N00014-93-1-0686, NSF under Grant No. MIP-9502885, and the ONR under Grant No. N00014-96-1-0930.

this sense, the available diversity manifests itself in the form of intersymbol interference, and the equalizer is the mechanism by which such diversity is actually exploited.

In a similar manner, spread-signature CDMA systems [1] and spread-response precoding systems [2], in which all users spread the transmission of their symbols in time, are designed to enable temporal diversity to be exploited. In this case, too, a suitably designed equalizer allows this diversity to be exploited. Finally, the use of antenna arrays—either at the receiver [3] or transmitter [4]—in conjunction with suitably designed equalizers allows allows spatial diversity to be exploited.

There is a substantial literature on the problem of multiple-access interference cancellation in such systems, and a wide range of efficient algorithms have been proposed for use in receivers—see, e.g., [5] [6] [7] [8] [9]. However, in such problems it is generally assumed that the fading in the channel is flat over the total bandwidth and over the symbol duration. By contrast, there have been comparatively few results presented on multiple-access interference cancellation for use on channels without such flat fading. In this work we develop efficient linear minimum mean-square error (MMSE) equalizer structures for such scenarios, which jointly suppress both multiple-access and intersymbol interference and therefore exploit the inherent diversity. In the case, of flat fading, the resulting algorithms yield attractive recursive implementations of some familiar multiuser detection algorithms.

2 System Model and Problem Formulation

Consider a passband CDMA system in which there are M users, all sharing a total fixed bandwidth LW_0, so that $W_0 L/M$ is the effective bandwidth per user. In the equivalent discrete-time baseband model for the system, the modulation process can be viewed as follows. The coded symbol stream of the mth user ($1 \leq m \leq M$), which we denote by $x_m[n]$, is modulated onto a unique signature sequence $h_m[n]$ to produce $y_m[n]$ which is transmitted within the total available bandwidth.

Conceptually, it is convenient to view the modulation process in two stages. As depicted in Fig. 1, these stages correspond to upsampling (i.e., zero-insertion) by a factor $L \geq M$, followed by linear time-invariant filtering with the signature sequence, i.e.,

$$y_m[n] = \sum_k x_m[k]\, h_m[n - kL]. \tag{1}$$

We emphasize at the outset that we avoid imposing any constraint on the length K of the signature sequences in this system. In so doing, our results will apply not only to conventional CDMA systems, for which $K = L$, but to recently proposed spread-signature CDMA systems for which $K \gg L$ to allow the temporal diversity benefit to be realized in the presence of time-selective fading.

Often, but certainly not always, the signatures are chosen to satisfy some convenient orthogonal properties. For example, often the $h_m[n]$ are chosen so that

$$\sum_k \mathbf{h}[k - nL]\, \mathbf{h}^{\mathrm{T}}[k - mL] = \delta[n - m]\, \mathbf{I} \tag{2}$$

where

$$\mathbf{h}[n] = \begin{bmatrix} h_1[n] & h_2[n] & \cdots & h_M[n] \end{bmatrix}^{\mathrm{T}}. \tag{3}$$

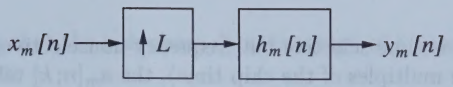


Figure 1: *Modulation of the the mth user's coded symbol stream* $x_m[n]$ *onto a signature sequence* $h_m[n]$ *for transmission.*

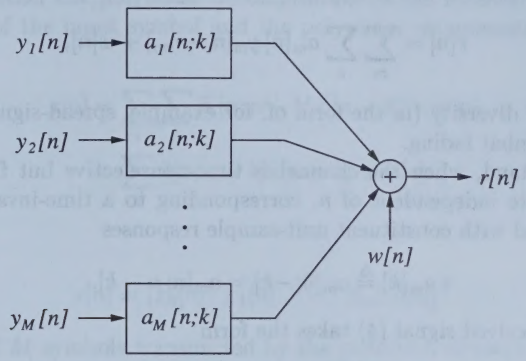


Figure 2: *General multiuser intersymbol interference channel model, where* $a_m[n]$ *denotes the unit-sample response corresponding to the mth user.*

Note that most familiar multiple-access techniques fit into this framework—i.e., in addition to CDMA, both time-division multiple-access (TDMA) and frequency-division multiple-access (FDMA) systems have implementations of the form depicted in Fig. 1 and have signatures satisfying (2). We emphasize, however, that we will *not* assume that (2) is satisfied in our development. In fact, in many cases the users can share the same signature without impacting performance; in such cases, users are distinguished by the channels their respective transmissions pass through. In any event, we will assume that the signature sequences are all known at the receiver.

The multiuser channel we consider, which is depicted in Fig. 2, corresponds to a rather general intersymbol interference environment. In this model, the complex-valued and, in general, time-varying channel response experienced by the mth user's transmission is $a_m[n;k]$, the response at time n to a unit sample input at time k. Hence, the sequence obtained at the receiver is

$$r[n] = \sum_m \sum_k a_m[n;n-k]\, y_m[n-k] + w[n] \tag{4}$$

where $w[n]$ is a zero-mean, complex-valued stationary circular white Gaussian sequence with variance

$$E\left[|w[n]|^2\right] = \mathcal{N}_0. \tag{5}$$

The channel responses $a_m[n;k]$ take into account both the physical propagation medium and path losses (amplitude variations) as well as relatively delays among the users transmissions due to the inherent asynchrony in the system. We will frequency restrict our attention to the case in which the $a_m[n;k]$ are causal, so that $a_m[n;k] = 0$ for $k > n$, and finite length, i.e., $a_m[n;k] = 0$ for $k \leq n - N$.

When the channel is time-selective but frequency-nonselective, corresponding to flat fading (and delays are multiples of the chip time), the $a_m[n;k]$ take the form

$$a_m[n;k] = a_m[n]\,\delta[n-k-n_m] \tag{6}$$

where n_m is the associated delay. In this case, the received signal (4) specializes to

$$r[n] = \sum_m \sum_k a_m[n]\, y_m[n-n_m] + w[n], \tag{7}$$

and only temporal diversity (in the form of, for example, spread-signature CDMA) can be exploited to combat fading.

On the other hand, when the channel is time-nonselective but frequency-selective, the $a_m[n;n-k]$ are independent of n, corresponding to a time-invariant, intersymbol interference channel with constituent unit-sample responses

$$a_m[k] \triangleq a_m[0;-k] = a_m[n;n-k]. \tag{8}$$

In this case, the received signal (4) takes the form

$$r[n] = \sum_m \sum_k a_m[k]\, y_m[n-k] + w[n], \tag{9}$$

and only spectral diversity can be exploited to combat fading.

In the sequel, it will be convenient to combine the signature modulation process with the effects of the channel to obtain an equivalent model in which the symbol streams of the individual users are time-division multiplexed before transmission over a multiuser channel where the unit-sample responses are now

$$\tilde{a}_m[n;k] = \sum_l a_m[n;l]\, h_m[l-k], \tag{10}$$

which in the time-invariant case specializes to

$$\tilde{a}_m[n] = a_m[n] * h_m[n]. \tag{11}$$

The multiuser equalization problem we consider is then one of recovering MMSE estimates $\hat{x}_m[n]$ of the symbol streams of the constituent users from a received signal $r[n]$ of the general form (4).

2.1 Polyphase Decompositions and State-Space Models

In developing our equalizers for the multiuser channel, it is convenient to express the received signal as an observation of the state of a multiple-input-multiple-output linear system. Because of the upsampling inherent in the modulation process, the notion of a polyphase decomposition of a signal will be very useful. The Lth order polyphase decomposition of an arbitrary signal $p[n]$ is the vector of sequences

$$\mathbf{p}[n] \triangleq \begin{bmatrix} p[nL] & p[nL+1] & \cdots & p[nL+L-1] \end{bmatrix}^{\mathrm{T}}.$$

The polyphase decomposition of a general time-varying channel response $q[n;k]$ is the vector of kernels

$$\mathbf{q}[n;k] \triangleq \begin{bmatrix} q[nL;kL] & q[nL+1;kL] & \cdots & q[nL+L-1;kL] \end{bmatrix}^{\mathrm{T}}. \tag{12}$$

Note that (12) implies that the polyphase decomposition of a specifically time-invariant kernel $q[n]$ has the form

$$\mathbf{q}[n] = \begin{bmatrix} q[nL] & q[nL+1] & \cdots & q[nL+L-1] \end{bmatrix}^{\mathrm{T}}.$$

With this notation the polyphase decomposition of the received signal can be expressed in terms of the input symbol and the polyphase components of the composite channels

$$\begin{aligned} \mathbf{r}[n] &= \sum_i \sum_k \tilde{\mathbf{a}}_i[n; n-k] x_i[n-k] + \mathbf{w}[n] \\ &= \sum_k \tilde{\mathbf{a}}[n; n-k] \mathbf{x}[n-k] + \mathbf{w}[n] \end{aligned} \tag{13}$$

where

$$\mathbf{x}[n] = \begin{bmatrix} x_0[n] & x_1[n] & \cdots & x_{M-1}[n] \end{bmatrix}^{\mathrm{T}}$$

is the collection of M symbols transmitted by the collection of users at time n, $\mathbf{w}[n]$ is the polyphase representation for the receiver noise $w[n]$, and

$$\tilde{\mathbf{a}}[n; k] = \begin{bmatrix} \tilde{\mathbf{a}}_0[n; k] & \tilde{\mathbf{a}}_1[n; k] & \cdots & \tilde{\mathbf{a}}_{M-1}[n; k] \end{bmatrix} \tag{14}$$

is the collection of polyphase decompositions of the channel responses $a_m[n; k]$, $m = 0, 1, \ldots, M-1$.

We can write the polyphase decomposition of the received signal as the output of a linear dynamical system of the form

$$\mathbf{s}[n+1] = \mathbf{F}\,\mathbf{s}[n] + \mathbf{G}\,\mathbf{x}[n+1] \tag{15a}$$

$$\mathbf{r}[n] = \mathbf{A}[n]\,\mathbf{s}[n] + \mathbf{w}[n], \tag{15b}$$

where the $\tilde{K}M$-dimensional state $\mathbf{s}[n]$ is of the form

$$\mathbf{s}[n] = \begin{bmatrix} \mathbf{x}[n] \\ \mathbf{x}[n-1] \\ \vdots \\ \mathbf{x}[n-\tilde{K}+1] \end{bmatrix} \tag{16}$$

with

$$\tilde{K} = \left\lceil \frac{K+N-1}{L} \right\rceil \tag{17}$$

denoting the effective length of the polyphase components of the time-varying channel responses $\tilde{a}_m[n; k]$. In the state equation (15a), we therefore have

$$\mathbf{F} = \begin{bmatrix} \mathbf{0} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{I} & \mathbf{0} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{I} & \mathbf{0} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \mathbf{0} & \mathbf{0} & \cdots & \mathbf{I} & \mathbf{0} \end{bmatrix}, \qquad \mathbf{G} = \begin{bmatrix} \mathbf{I} \\ \mathbf{0} \\ \mathbf{0} \\ \vdots \\ \mathbf{0} \end{bmatrix},$$

while in the observation equation (15b) we have that $\mathbf{A}[n]$ is the following $L \times \tilde{K}M$ matrix containing the channel and signature information

$$\mathbf{A}[n] = \left[\tilde{\mathbf{a}}[n;n] \;\; \tilde{\mathbf{a}}[n;n-1] \;\; \cdots \;\; \tilde{\mathbf{a}}[n;n-\tilde{K}+1]\right]$$

We assume that the users are uncorrelated and transmit white symbol streams, i.e.,

$$E\left[\mathbf{x}[n]\mathbf{x}^{\dagger}[m]\right] = \mathcal{E}_s \mathbf{I}\,\delta[n-m].$$

Note that in the time-invariant channel case, the observation matrix $\mathbf{A}[n]$ is a constant $\mathbf{A}$, i.e., independent of n. We also note that a state space model of the form (15) also applies when in addition there are multiple antennas at the transmitters and receiver in the system. For these extensions, state and/or observation vectors of appropriately higher dimension are used.

3 MMSE Multiuser Equalizers

A linear MMSE equalizer for the multiple-access system described by (15) necessarily requires an estimate of the state (16). Such an estimate can be computed recursively and efficiently via the Kalman filtering algorithm. Denoting by $\hat{\mathbf{s}}[n|k]$ the MMSE estimate of the state at time n given observations of $\mathbf{r}[l]$ up to time k—and denoting the associated error covariance by $\mathbf{\Lambda}[n|k]$, the state estimation (equalizer) equations take the form [10] [11]

$$\hat{\mathbf{s}}[k|k] = \mathbf{F}\hat{\mathbf{s}}[k-1|k-1] + \boldsymbol{\mu}[k]\left(\mathbf{r}[k] - \mathbf{A}[k]\mathbf{F}\hat{\mathbf{s}}[k-1|k-1]\right) \tag{18a}$$

$$\boldsymbol{\mu}[k] = \mathbf{\Lambda}[k|k-1]\mathbf{A}^{\dagger}[k]\left(\mathbf{A}[k]\mathbf{\Lambda}[k|k-1]\mathbf{A}^{\dagger}[k] + \mathcal{N}_0\mathbf{I}\right)^{-1} \tag{18b}$$

$$\mathbf{\Lambda}[k|k] = \left(\mathbf{I} - \boldsymbol{\mu}[k]\mathbf{A}[k]\right)\mathbf{\Lambda}[k|k-1] \tag{18c}$$

$$\mathbf{\Lambda}[k+1|k] = \mathbf{F}\mathbf{\Lambda}[k|k]\mathbf{F}^{\mathrm{T}} + \mathcal{E}_s\mathbf{G}\mathbf{G}^{\mathrm{T}} \tag{18d}$$

and are initialized with $\hat{\mathbf{s}}[-1|-1] = \mathbf{0}$ and $\mathbf{\Lambda}[-1|-1] = \mathbf{0}$. Note that although we do not need to compute it explicitly, the one-step prediction is given by

$$\hat{\mathbf{s}}[k+1|k] = \mathbf{F}\,\hat{\mathbf{s}}[k|k]. \tag{18e}$$

There are several notable features of these estimation equations. First, because of the form of the state vector (16) smoothed estimates of the symbols $\mathbf{x}[n-k]$ for $1 \le k \le \tilde{K}-1$ are available at the same time as a filtered estimate of $\mathbf{x}[n]$ is computed. In particular, our estimates of the symbol streams of the users are given by

$$\hat{x}[n-k|n] = \mathbf{B}[k]\,\hat{\mathbf{s}}[n|n], \qquad 0 \le k \le \tilde{K}-1 \tag{19}$$

where

$$\mathbf{B}[k] = \left[\mathbf{B}_0[k] \;\; \mathbf{B}_1[k] \;\; \cdots \;\; \mathbf{B}_{\tilde{K}-1}[k]\right] \tag{20}$$

with

$$\mathbf{B}_j[k] = \begin{cases} \mathbf{I} & j = k \\ \mathbf{0} & \text{otherwise} \end{cases} \tag{21}$$

These smoothed estimates are available, in effect, for free—no additional computation is required.

We also remark that further smoothed estimates can be obtained by augmenting the state vector (16) with additional lags of the transmitted symbols, i.e.,

$$\mathbf{s}'[n] = \begin{bmatrix} \mathbf{s}[n] \\ \mathbf{x}[n-\tilde{K}] \\ \mathbf{x}[n-\tilde{K}-1] \\ \vdots \\ \mathbf{x}[n-\tilde{K}'+1], \end{bmatrix}. \tag{22}$$

and modifying the state equations (15) by augmenting the matrices $\mathbf{F}$ and $\mathbf{A}[n]$ to obtain $\mathbf{F}'$ and $\mathbf{A}'[n]$, respectively. In practice, computationally more efficient algorithms can be used to obtain these additional smoothed estimates.

Note, too, that the update equation for the state estimate (18a) consists of two parts. The first part is a prediction of the state based on the state estimate at the previous symbol time. The second part is a correction term based on the difference between the prediction and most recently received observation. When the data symbols are discrete-valued (and uncoded), performance can often be enhanced by replacing $\hat{\mathbf{s}}[k-1|k-1]$ with the associated decisions obtained by thresholding. For example, if $x_m[n] \in \{-1, +1\}$, then $\hat{\mathbf{s}}[k-1|k-1]$ can be replaced with $\mathrm{sgn}(\hat{\mathbf{s}}[k-1|k-1])$. The result is an attractive decision-feedback equalizer structure.

Finally, note that the algorithm requires that the covariance of the estimation error $\mathbf{\Lambda}[k|k]$ be computed at each time step. This reliability information can potentially be used to enhance the performance of a higher level error correction scheme (involving, for example, soft-decision decoding).

Other related equalizer structures can also be related to (18). For example, a recursive implementation of the zero-forcing multiuser equalizer is obtained by replacing the gain (18b) with

$$\boldsymbol{\mu}[k] = \mathbf{\Lambda}[k|k-1]\mathbf{A}^{\dagger}[k]\left(\mathbf{A}[k]\mathbf{\Lambda}[k|k-1]\mathbf{A}^{\dagger}[k]\right)^{-1}. \tag{23}$$

Similarly, a recursive implementation of the matched-filter multiuser equalizer is obtained using the gain

$$\boldsymbol{\mu}[k] = \mathbf{\Lambda}[k|k-1]\mathbf{A}^{\dagger}[k]/\mathcal{N}_0. \tag{24}$$

It is important to emphasize, however, that in these cases the matrices $\mathbf{\Lambda}[k|k]$ and $\mathbf{\Lambda}[k+1|k]$ no longer correspond to the associated estimation errors, except for limiting cases: $\mathcal{N}_0 \to 0$ in the zero-forcing case, and $\mathcal{N}_0 \to \infty$ in the matched-filter case.

3.1 Near-Far Resistance

It is straightforward to verify that given some mild conditions on the channel the MMSE multiuser equalizer is near-far resistant—i.e., that $\mathbf{\Lambda}[k|k] \to \mathbf{0}$ for all $k \geq 0$ when $\mathcal{N}_0 \to 0$. Note that since $\mathbf{\Lambda}[-1|-1] = \mathbf{0}$ it suffices to show that $\mathbf{\Lambda}[k+1|k+1] = \mathbf{0}$ if $\mathbf{\Lambda}[k|k] = \mathbf{0}$ and $\mathcal{N}_0 = 0$, and then apply induction. We consider the worst-case scenario corresponding to the maximum number of users, i.e., $M = L$.

From (18d), we obtain $\mathbf{\Lambda}[k+1|k] = \mathcal{E}_s \mathbf{G}\mathbf{G}^{\mathrm{T}}$. Hence it suffices to verify that this implies the righthand side of (18c) is $\mathbf{0}$. Letting

$$\mathbf{C}[k] = \mathbf{A}[k]\mathbf{G}, \tag{25}$$

we see that we need only show that

$$\mathbf{C}^{\dagger}[k] \left(\mathbf{C}[k]\mathbf{C}^{\dagger}[k]\right)^{-1} \mathbf{C}[k] = \mathbf{I}. \tag{26}$$

However, from (25) we see that $\mathbf{C}[k]$ is the $M \times M$ matrix

$$\mathbf{C}[k] = \tilde{\mathbf{a}}[k;k]. \tag{27}$$

Assuming (27) is invertible for each k then (26) follows immediately. Note that in the time-invariant case, $\mathbf{C}[k]$ is independent of k and takes the form

$$\mathbf{C} = \tilde{\mathbf{a}}[0] = \begin{bmatrix} \tilde{a}_0[0] & \tilde{a}_1[0] & \cdots & \tilde{a}_{M-1}[0] \\ \tilde{a}_0[1] & \tilde{a}_1[1] & \cdots & \tilde{a}_{M-1}[1] \\ \vdots & \vdots & \ddots & \vdots \\ \tilde{a}_0[L-1] & \tilde{a}_1[L-1] & \cdots & \tilde{a}_{M-1}[L-1] \end{bmatrix}. \tag{28}$$

3.2 Steady-State Equalization Error

When the channel is time-invariant and the equalization algorithm is operating in the steady-state, the state update equations reduce to

$$\hat{\mathbf{s}}[k|k] = \mathbf{F}\hat{\mathbf{s}}[k-1|k-1] + \boldsymbol{\mu}\left(\mathbf{r}[k] - \mathbf{A}\mathbf{F}\hat{\mathbf{s}}[k-1|k-1]\right) \tag{29a}$$

$$\boldsymbol{\mu} = \mathbf{\Sigma}_{\infty}\mathbf{A}^{\dagger}\left(\mathbf{A}\mathbf{\Sigma}_{\infty}\mathbf{A}^{\dagger} + \mathcal{N}_0\mathbf{I}\right)^{-1} \tag{29b}$$

where

$$\mathbf{\Sigma}_{\infty} = \lim_{k\to\infty} \mathbf{\Lambda}[k+1|k]$$

is steady-state prediction error covariance. This covariance is obtained as the solution to the Ricatti equation

$$\mathbf{\Sigma}_{\infty} = \mathbf{F}(\mathbf{\Sigma}_{\infty}^{-1} + \mathcal{N}_0^{-1}\mathbf{A}^{\dagger}\mathbf{A})^{-1}\mathbf{F}^{\mathrm{T}} + \mathcal{E}_s\mathbf{G}\mathbf{G}^{\mathrm{T}}, \tag{30}$$

which can be obtained using any of a number of algorithms—see, e.g., [10]. The associated steady-state filtering error covariance

$$\mathbf{\Lambda}_{\infty} = \lim_{k\to\infty} \mathbf{\Lambda}[k|k] \tag{31}$$

takes the form

$$\mathbf{\Lambda}_{\infty} = (\mathbf{I} - \boldsymbol{\mu}\mathbf{A})\,\mathbf{\Sigma}_{\infty} = \mathbf{\Sigma}_{\infty} - \mathbf{\Sigma}_{\infty}\mathbf{A}^{\dagger}\left(\mathbf{A}\mathbf{\Sigma}_{\infty}\mathbf{A}^{\dagger} + \mathcal{N}_0\mathbf{I}\right)^{-1}\mathbf{A}\mathbf{\Sigma}_{\infty} = \mathcal{N}_0\left(\mathcal{N}_0\mathbf{\Sigma}_{\infty}^{-1} + \mathbf{A}^{\dagger}\mathbf{A}\right)^{-1}. \tag{32}$$

where the second equality in (32) follows from (29b), and the third from an application of the matrix inversion lemma [11]. Expressions for the associated steady-state infinite-interval smoothing error covariance follow from the multichannel Wiener filtering results presented in [12].

3.3 Example

As an illustration of the performance of the algorithm, consider a simple example involving $M = 2$ users using conventional CDMA signatures of length $K = L = 4$. The signatures are orthogonal Hadamard sequences, and the two users transmit and equal power $\mathcal{E}_s$. The channels experienced by the two users are time-invariant and have real-valued unit-sample responses given by

$n =$	0	1	2	3	4
$a_0[n]$	0.328	0.176	0.011	0.099	-0.196
$a_1[n]$	0.678	0.024	0.719	0.106	0.349

The received powers are therefore quite asymmetric:

$$\mathcal{E}_s \sum_n |a_0[n]|^2 = 0.187\mathcal{E}_s, \qquad \mathcal{E}_s \sum_n |a_1[n]|^2 = 1.110\mathcal{E}_s. \tag{33}$$

The resulting equalization performance is depicted in Fig. 3. The top figures depict the frequency responses of the channels for each of the two users. The bottom left figure shows the combined mean-square equalization error in the signals for the two users after filtering. The near-far resistance of the equalization is apparent—there is no interference-limited behavior at high SNR. The bottom right figure shows the additional reduction in mean-square equalization error that can be obtained through the use of smoothing. That there is comparatively little benefit to such smoothing beyond a couple lags is a consequence of the comparatively short channel responses involved.

References

[1] G. W. Wornell, "Spread-signature CDMA: Efficient multiuser communication in the presence of fading," *IEEE Trans. Inform. Theory*, vol. 41, pp. 1418–1438, Sept. 1995.

[2] G. W. Wornell, "Spread-response precoding for communication over fading channels," *IEEE Trans. Inform. Theory*, vol. 42, pp. 488–501, Mar. 1996.

[3] J. G. Proakis, *Digital Communications*. New York, NY: McGraw-Hill, 2nd ed., 1989.

[4] G. W. Wornell and M. D. Trott, "Efficient signal processing techniques for exploiting transmit diversity on fading channels," *IEEE Trans. Signal Processing*, Jan. 1997. To appear, Special Issue on Signal Processing for Advanced Communications.

[5] S. Verdú, "Minimum probability of error for asynchronous Gaussian multiple-access channels," *IEEE Trans. Inform. Theory*, vol. IT-32, pp. 85–96, Jan. 1986.

[6] R. Lupas and S. Verdú, "Linear multiuser detectors for synchronous code-division multiple-access channels," *IEEE Trans. Inform. Theory*, vol. 35, pp. 123–136, Jan. 1989.

[7] R. Lupas and S. Verdú, "Near-far resistance of multiuser detectors in asynchronous channels," *IEEE Trans. Commun.*, vol. 38, pp. 496–508, Apr. 1990.

[8] U. Madhow and M. L. Honig, "MMSE interference suppression for direct-sequence spread-spectrum CDMA," *IEEE Trans. Commun.*, vol. 42, pp. 3178–3188, Dec. 1994.

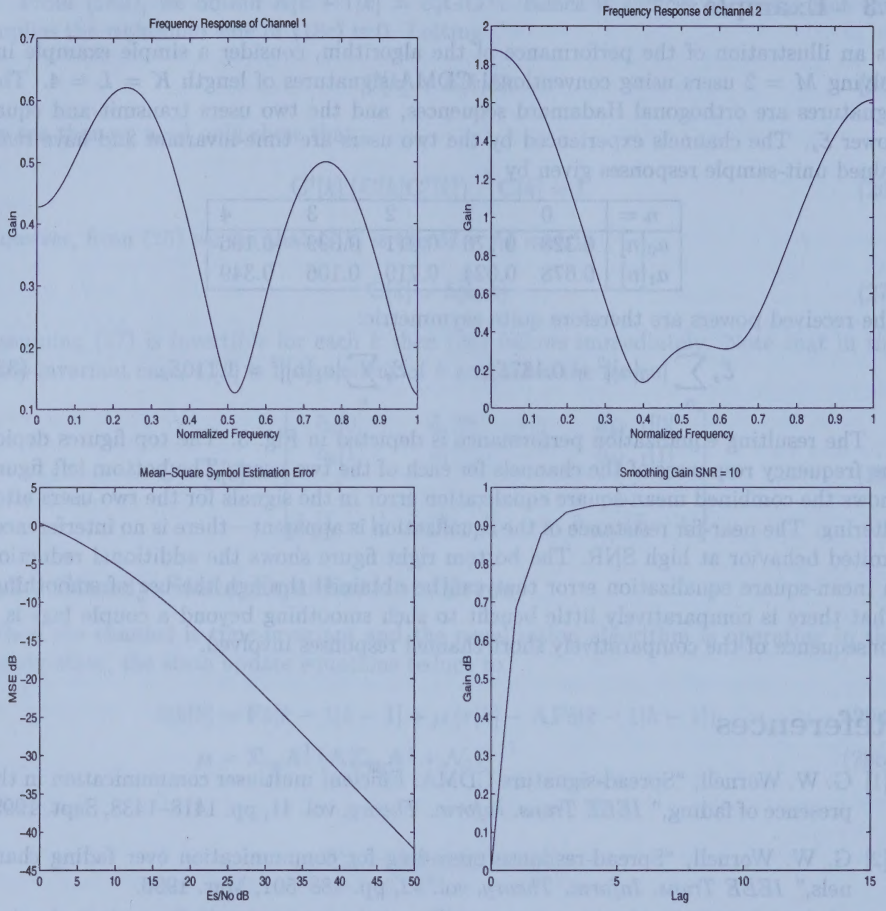


Figure 3: *Performance of the linear MMSE multiuser equalizer in a $M = 2$ user time-invariant scenario.*

[9] M. Honig, U. Madhow, and S. Verdú, "Blind adaptive multiuser detection," *IEEE Trans. Inform. Theory*, vol. 41, pp. 944–960, July 1995.

[10] B. D. O. Anderson and J. B. Moore, *Optimal Filtering.* Englewood Cliffs, NJ: Prentice-Hall, 1979.

[11] S. Haykin, *Adaptive Filter Theory.* Englewood Cliffs, NJ: Prentice-Hall, 1986.

[12] A. Duel-Hallen, "Equalizers for multiple input/multiple output channels and PAM systems with cyclostationary input sequences," *IEEE J. Select. Areas Commun.*, vol. 10, pp. 630–639, Apr. 1992.

Design and Implementation Issues for a Wideband, Indoor, DS-CDMA System Providing Multimedia Access

Craig Teuscher, Samuel Sheng, Ian O'Donnell, Kevin Stone, Robert Brodersen
Department of Electrical Engineering and Computer Science
University of California, Berkeley
Berkeley, CA 94720, USA

Abstract: This paper considers design and implementation issues for an indoor wireless system providing multimedia access to portable terminals. Power consumption is a key metric in this design, because of portability constraints. An integrated CMOS receiver implementation is presented which achieves total power consumption of less than 150 mW for a 1 Mbps data rate. An improved second generation system design is also considered.

1. Introduction

Wireless communication systems are expected to play a major role in providing portable access to future information services. Design of these systems presents key technical challenges: multimedia access requires high-bandwidth, low-latency network connections to many users; mobility requires adaptation to time-varying channel conditions; and portability imposes severe constraints on receiver size and power consumption. Most wireless research in this area has focused on the bandwidth and mobility issues; power considerations have been largely ignored.

This paper proposes a more unified approach to the portable receiver design problem. Experience has shown that system level design choices have enormous impact on overall power consumption [1]. Consequently, low power implementation requires a methodology that explicitly considers power in all stages of the design process. One such methodology is presented here.

The InfoPad project at UC Berkeley provides a demonstration vehicle for this methodology [3]. The system is designed to reflect future wireless system requirements. A conceptual illustration is shown in Figure 1. The goal is to provide wireless multimedia access to as many as 50 portable terminals at data rates up to 1 Mbit/sec per terminal. Portable receiver power consumption is limited to approximately 250 mW.

A direct-sequence code division multiple access (DS-CDMA) system design has been proposed for this application [3]. A first generation chipset has been implemented to realize the design, and to serve as a starting point for future investigations [4]. The custom chipset achieves target power consumption levels, but provides marginal link performance in the presence of many users. An improved system design is under development which incorporates the same low power design methodology, but improves system performance using enhanced receiver techniques.

Organization of the paper is as follows: section 2 provides a brief overview of the system; section 3 details the low power design methodology; section 4 discusses the integrated CMOS receiver implementation; and section 5 considers the improved second generation design.

2. System Overview

The InfoPad wireless communication system is characterized by the following distinctive features:

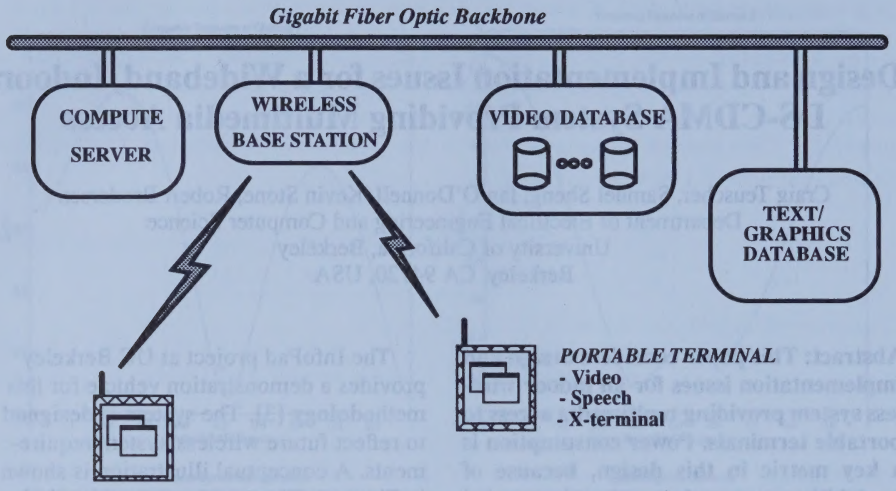


Figure 1. Conceptual diagram of the InfoPad system for wireless multimedia access.

- **asymmetric links**: uplink and downlink bandwidth requirements are asymmetric. The downlink (base station to portable receiver) transports audio, full-motion video, text, and graphics information, requiring data rates up to 1 Mbps per user. The uplink, in contrast, conveys user commands (consisting of pen strokes and voice commands) to the network, and requires peak bandwidths less than 64 kbps per user. For purposes of this paper, only the downlink design is considered.
- **low power requirements**: power consumption constitutes a critical design metric in the portable terminal in order to extend battery life. Target power consumption for the entire terminal is less than 2W. This results in a power budget of approximately 250 mW for the portable receiver subsystem.
- **indoor operation**: system design assumes an indoor, office-type environment. Consequently, indoor multipath and shadowing effects constitute primary sources of performance degradation on the link. Picocells and sectored antennas are used to reduce base station transmit power levels, and to minimize intercell interference.
- **quasi-static channel**: most of the time, the portable terminal is stationary. When the environment does change (e.g. people move within a room, or the terminal itself is moved) any motion occurs at human rates. Consequently, the coherence time of the channel is assumed to be much greater than a symbol period. This permits accurate pilot channel estimates at the portable receiver, and allows Doppler effects to be ignored.
- **synchronous cell transmission**: all data destined for users within a particular cell emanates from a single network-attached base station. As a result, user data within a single cell is easily synchronized. At a particular portable receiver, all users' signals experience an identical dispersive channel transfer function.
- **error tolerance**: wireless links require a system design which tolerates higher error rates than wired connections. Error-resilient coding and compression algorithms are used at the base station to reduce retransmission overhead. The InfoPad system design anticipates bit error rates (BER) up to 10^{-2} under heavy user loading conditions.

For this application, a DS-CDMA system design was selected primarily based on the power considerations discussed below. In addition, CDMA mitigates the effects of multipath channel dispersion and reduces the probability that the entire signal is lost in a local deep fade. From a capacity perspective, CDMA is particularly attractive on the downlink, since user signals can be synchronized.

3. A Low Power Design Methodology [1]

Low power receiver design requires a methodology that explicitly considers power consumption in all stages of the design process. System, algorithmic, architectural, and circuit level optimizations are required to achieve significant power reduction. Although system and architectural issues are the focus of this paper, a global approach is implicit.

Key components of the methodology include: digitizing the signal at the earliest possible stage; simplifying the remaining analog hardware requirements; and leveraging low power architectural techniques in the digital section. CDMA receiver design is well-suited to this approach.

From a system perspective, location of the analog-to-digital (A/D) conversion constitutes one of the most important design choices influencing overall power dissipation. Because digital designs are particularly amenable to low power techniques, it is desirable to "go digital" at the earliest possible point in the receive chain. This leads to the mostly digital receiver architecture shown in Figure 2. Analog blocks are used only for filtering, amplification, demodulation, sampling, and A/D conversion. All other functions are implemented digitally at baseband.

Analog blocks which remain in the system are further simplified to reduce complexity. In a DS-CDMA system, three important simplifications are possible:

1. **VGA dynamic range reduction:** spreading the signal wider than the coherence bandwidth of the channel reduces the likelihood that the entire signal is lost in a local deep fade. This, in turn, relaxes the dynamic range requirements for the variable gain amplifier (VGA).
2. **Sampling demodulation:** demodulation and sampling can be combined into a single step [3]. A carefully controlled passive sampling switch replaces the standard mixer-sampler configuration. No PLL is required, yielding significant power savings at the expense of increased phase noise.
3. **A/D resolution reduction:** because the signal is quantized prior to correlation, precise quantization is not required. Software simulation was used to determine the appropriate A/D converter resolution. Figure 3 illustrates the results for a typi-

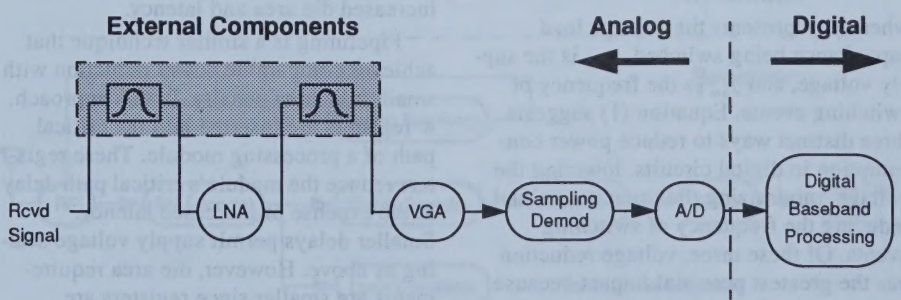


Figure 2. Mostly digital receiver architecture.

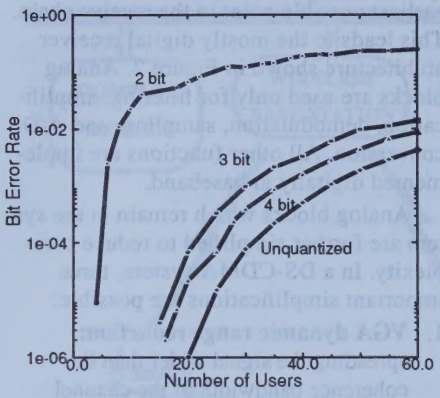


Figure 3. Simulated bit error rate vs. number of users for varying quantizer resolutions.

cal channel. For resolutions greater than four bits, multiple access interference (MAI) dominates the quantization noise, yielding little performance increase at higher levels of quantization. Limiting the converter to four bit resolution has two desirable effects: first, power consumption in the A/D converter is reduced dramatically; second, subsequent digital processing requirements are minimized.

Once the signal is converted to a digital format, an important class of power reduction techniques is available. The classic equation governing dynamic power dissipation in a digital circuit is:

$$P = C_L V_{dd}^2 f_{sw} \qquad (1)$$

where C_L represents the average load capacitance being switched, V_{dd} is the supply voltage, and f_{sw} is the frequency of switching events. Equation (1) suggests three distinct ways to reduce power consumption in digital circuits: lowering the voltage, minimizing the capacitance, and reducing the frequency of switching events. Of these three, voltage reduction has the greatest potential impact because of the quadratic dependence expressed above.

Of course, voltage reduction alone is not a viable strategy because of throughput considerations. Lowering the supply voltage increases propagation delays, thereby decreasing circuit speeds. To maintain fixed throughput, architectural design changes are necessary to compensate for the effects of low voltage operation. Two of the most important architectural techniques are parallel processing and pipelining.

Parallel processing relies on hardware duplication to permit supply voltage scaling. Consider a datapath module operating at a frequency f while switching an average capacitance C at a supply voltage of V. The average power consumption is

$$P_0 = CV^2 f \qquad (2)$$

If the algorithm is parallelizable, N identical modules can be operated in parallel, each at a frequency $\frac{f}{N}$. Since the speed requirements for each module are reduced, the supply voltage can be lowered until gate delays increase by an offsetting amount. The new supply voltage is αV, where α represents the voltage reduction factor. Neglecting overhead, circuit capacitance increases by a factor of N. The overall power consumption is now

$$P = (NC)(\alpha V)^2\left(\frac{f}{N}\right) \qquad (3)$$
$$= \alpha^2 P_0$$

The cost of this power reduction is increased die area and latency.

Pipelining is a similar technique that achieves comparable power reduction with smaller die area penalty. In this approach, N registers are inserted into the critical path of a processing module. These registers reduce the module's critical path delay at the expense of increased latency. Smaller delays permit supply voltage scaling as above. However, die area requirements are smaller since registers are generally more compact than duplicate hardware. Pipelining can also be combined

with parallel processing to capture the benefits of both.

Additional power savings can be achieved by reducing capacitance or the frequency of switching events. Techniques include the use of minimum size devices, intelligent numeric representation for data (e.g. twos complement vs. sign magnitude), and power down of clock circuitry and associated logic when processing is not required. Other optimizations are considered in [2].

4. A CMOS Receiver Implementation

The low power design methodology has been applied to a hardware implementation of the DS-CDMA receiver [4]. A high-level block diagram of the first generation system is depicted in Figure 4. The design is based loosely on the US IS-95 digital cellular CDMA standard, but achieves a data rate more than an order of magnitude higher. A DQPSK modulation format encodes two bits per symbol at a symbol rate of 512K baud for each user. Orthogonal Walsh-Hadamard functions of length 64 are used as spreading codes with a chipping rate of 32 Mchips/sec. Six bits of power control resolution are provided to each user, and the composite data signal is then scrambled using a maximal-length shift register (MLSR) sequence. After filtering, the signal is modulated to a carrier frequency of 1.056 GHz prior to transmission.

At the receiver, the signal is filtered, amplified, demodulated, and sampled at 64 MHz (twice the chip rate). After four bit signal quantization, the PN scrambling sequence is removed, and a single Walsh code correlator selects the desired user's data. Timing recovery and pilot channel estimates are also provided.

A two chip receiver solution has been designed and fabricated using standard digital 0.8 micron CMOS technology. The

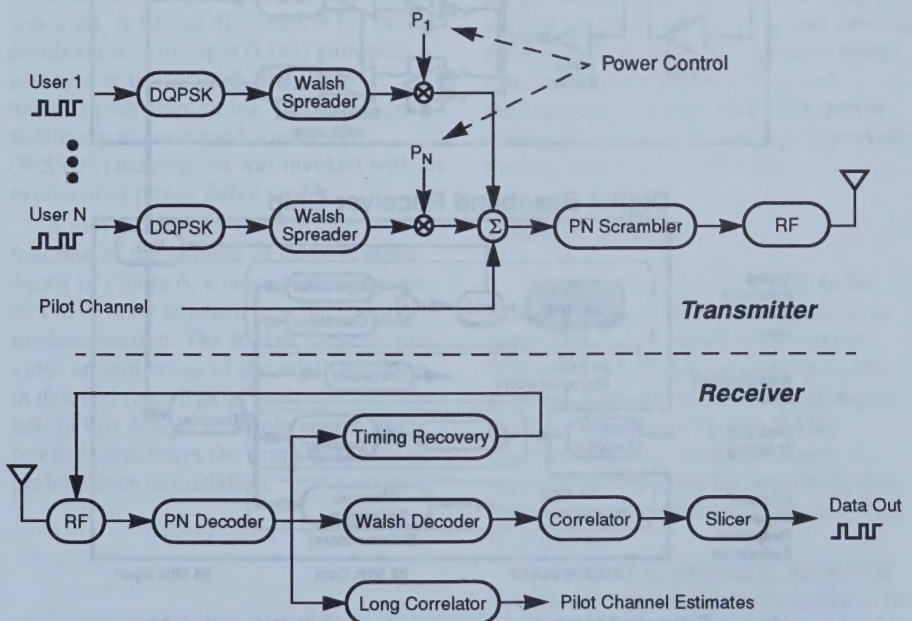


Figure 4. Transmitter and receiver block diagrams for the first generation system.

Analog Front-End Receive Chip	
Area:	6.2 mm x 6.2 mm
Transistor Count:	6500
Clock rate:	64 MHz
Power:	VGA: 66.4 mW (8.3 mW per SHA)
	A/D's: 4.4 mW (2.2 mW per A/D)
	RF LNA: 36 mW @3.3V
	TOTAL: 106.8 mW
VGA:	4 stages, 3 dB gain per stage
LNA:	11.0 dB peak S21 @ 900 MHz
Sampling Demodulator:	Tested to 1.1 GHz
	No measurable analog harmonic distortion below +10 dBm

Digital Baseband Receive Chip	
Area:	7.5mm x 7.4mm
Transistor Count:	87000
Clock rate:	Correlator: 32 MHz
Power/Supply:	3 supplies used
	4.2 mW @ 1.5V
	8.3 mW @ 3.3V
	14.9mW @ 5V
	TOTAL: 27.4 mW

Table 1: Performance Results for Analog and Digital Receive Chips.

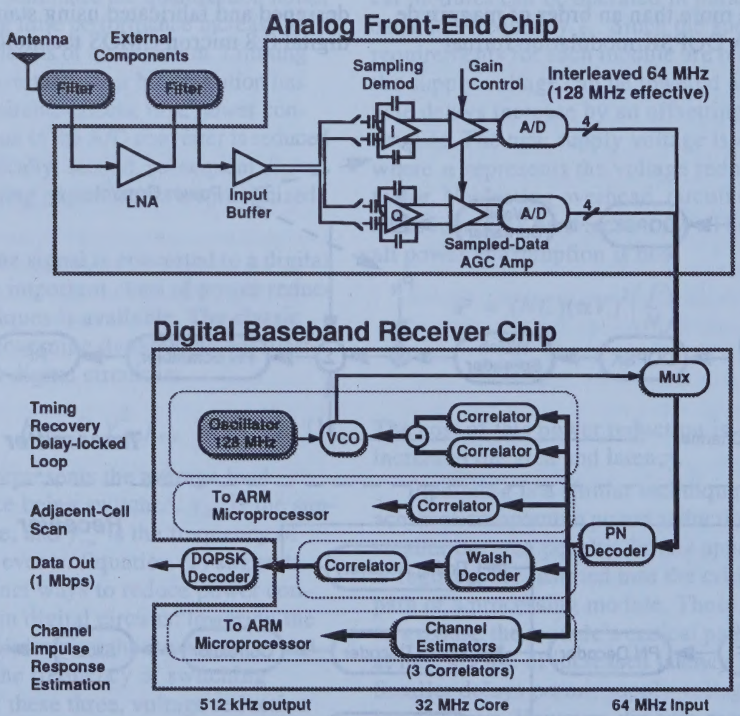


Figure 5 Detailed block diagrams of analog and digital receive chips.

first chip implements the analog portion of the receiver; the second includes all digital baseband processing. Detailed block diagrams for the two chips are shown in Figure 5. Table 1 summarizes the performance of the chipset. Total power consumption of the analog and digital chips are 107 mW and 27 mW, respectively.

From Figure 5 it is clear that the critical block in the digital section of the receiver is the complex valued correlator, since seven are required to operate at the full 32 MHz chip rate (two for timing recovery, three for pilot channel estimates, one for adjacent cell scan, and one for data recovery). By minimizing the critical path in the accumulator using a bit-level pipelined carry-save adder architecture, a supply voltage as low as 1.5V can be used. This results in a measured power consumption of 0.6 mW per correlator.

BER analysis of the first generation system design was presented in [5]. That work incorporated standard assumptions about the wideband indoor radio channel. A linear, time-invariant FIR channel model was used. A Rician distribution for the dominant line-of-sight (LOS) path was assumed with Rayleigh-distributed delayed path amplitudes. The wide-sense stationary uncorrelated scattering (WSSUS) assumption was invoked with an exponential power delay profile.

A plot of average bit error rate as a function of the number of users is reproduced in Figure 6, where system parameters have been adjusted to reflect actual implementation. The Rician K-factor provides an indication of the relative energy in delayed (interfering) paths for the channel. In this design, multiple access interference constitutes the primary source of performance degradation.

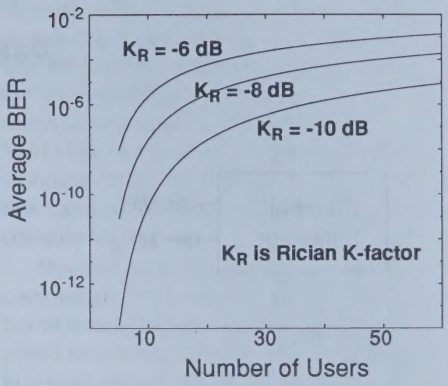


Figure 6. Average bit error rate vs. number of users. Parameters are T_d=50 ns, N=64, T_{chip}=31μs.

5. An Improved System Design

Performance results for the custom chipset validate the low power design methodology presented previously. The same methodology can be applied to more sophisticated receiver architectures, yielding higher performance receivers suitable for portable applications. In this vein, a second generation design is under development which incorporates diversity reception, improved code selection, and multiuser detection to boost link performance. Power consumption again provides the key metric for this design.

Diversity reception

Diversity reception is a well-established technique for improving receiver performance in a fading environment. Wideband DS-CDMA systems provide inherent diversity by resolving multipath signal components [7]. The RAKE receiver structure, shown in Figure 7, exploits this diversity by coherently combining delayed path signal energies prior to detection.

Because of its simplicity, the RAKE receiver structure is also well-suited to low power implementations. Each finger of the RAKE requires a single correlator operat-

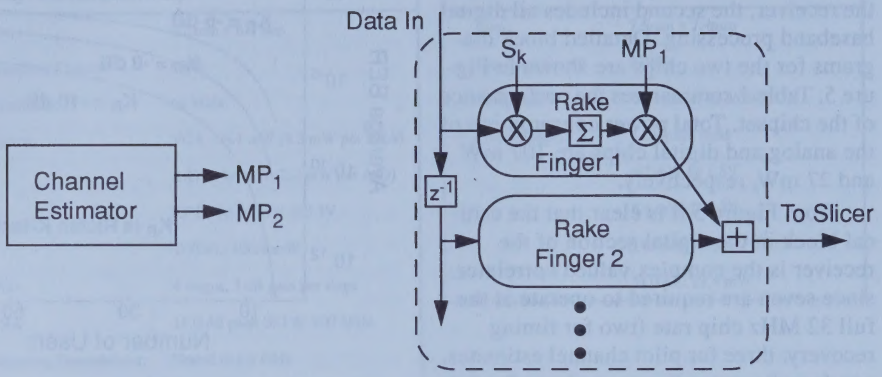


Figure 7. Structure of the RAKE Receiver.

ing in parallel with the original data correlator. Power dissipation for this extra element is less than 0.6 mW. Gain combining strategies are possible with a careful ordering of operations. Complex multiplications are shifted after correlations to minimize the frequency of switching events. Correlators still operate at the chip rate, but multipliers now require only symbol-rate throughput. Additional power savings can be achieved by "shutting down" RAKE fingers which do not capture significant signal energy. Pilot channel estimates provide a convenient way to monitor this condition.

Although the RAKE receiver provides diversity reception in a simple, low power implementation, its performance is severely degraded in the presence of strong MAI [16]. Consequently, effective RAKE operation requires a system design which eliminates this interference. For small numbers of users, this is achievable with appropriate user code selection; for larger numbers of users, multiuser detection is required.

Code Selection

User code selection can have a significant impact on performance and power consumption in the portable receiver. For the second generation system, the PN-Walsh overlay code is abandoned in favor of a coding scheme better suited to indoor channel characteristics, and to low power implementations.

In typical CDMA systems, codes are required to possess good auto- and cross-correlation properties for all time offsets [6]. As a result, Gold codes and PN-Walsh overlay codes are usually employed. For a synchronous system, however, user codes require low cross-correlation values *only for offsets smaller than the time dispersion of the channel.* This less restrictive condition allows sequences which achieve considerably smaller cross-correlation values at the specific offsets of interest.

For a wideband radio channel with transmit bandwidth B, the frequency selective channel can be represented using a tapped delay line model with tap spacing $1/B$. The number of resolvable paths $L = \lfloor BT_M \rfloor + 1$ where T_m is the channel multipath spread and $\lfloor x \rfloor$ is the largest integer less than or equal to x. Extensive measurements of the indoor channel under a variety of operating conditions have been reported [17,18,19,20]. Typical multipath spread values range from 25-75 ns, and exhibit an exponential power delay profile. In-vivo measurements are currently being collected with the InfoPad system to con-

firm these results. For a transmit bandwidth of 40 MHz, this yields a diversity order of $L \approx 3-4$. Intelligent code design requires mutually low cross-correlations only for time offsets smaller than this diversity order.

In the proposed system, user codes are constructed using distinct phase offsets of a single prime MLSR sequence. Let d_{offset} represent the minimum phase offset between any two user codes derived from the same sequence. In order to provide adequate multipath rejection, constrain $d_{offset} \geq L$. The first MLSR sequence thus generates at most $\frac{N}{L}$ user codes. The mutual cross-correlation $R_{ij}(k)$ between user codes i and j at phase offset k is given by

$$R_{ij}(k) = \begin{cases} 1 & i = j, k = 0 \\ 1 & i \neq j \\ & k = d_{offset}, 2d_{offset}, \ldots \\ -\frac{1}{N} & otherwise \end{cases} \quad (4)$$

where N is the code length. Additional user codes are generated in a similar fashion using other MLSR sequences. Once a second prime sequence is introduced, $R_{ij}(k)$ becomes three valued for $|i-j| > \frac{N}{L}$.

The advantage of this code set is that if the number of active users is less than $\frac{N}{L}$, signals at the receiver remain (approximately) mutually orthogonal *for all multipath delays of interest.* Orthogonality has important consequences for low power design. In the absence of MAI, a simple, single-user RAKE receiver provides optimal detection. This means that if the portable receiver can learn when the system is lightly-loaded, it can shift into a low power mode of operation by shutting down additional multiuser detection processing.

Multiuser detection

Once the number of active users exceeds $\frac{N}{L}$, multiple access interference can no longer be entirely eliminated through code selection; more sophisticated processing is required. Recent work on CDMA receiver design has demonstrated the shortcomings of simple, single-user receiver structures in the presence of MAI [16] Multiuser detection exploits the structure of this interference to increase link capacity at the expense of additional complexity.

Optimal multiuser detection requires a complexity that is exponential in the number of users [14]. Researchers have proposed numerous suboptimal techniques for practical systems [9,10,11,12,13,15]. Several of these proposals are currently being evaluated for incorporation into the second generation system design. The key metric in this evaluation is suitability for low power VLSI implementation.

In this context, the receiver architecture proposed in [15] appears especially attractive. The KL matched filter correlators are well-suited to a low power implementation and the decorrelating filter is easily parallelizable. BER performance for this receiver is substantially better than the single-user correlation receiver used in the first design. The multipath decorrelating detector eliminates the error probability floor and exhibits diversity gains even when the system is fully-loaded.

Unfortunately, like most multiuser detection techniques, this approach requires a substantial increase in quantizer resolution. Because multiuser detection eliminates MAI (usually at the expense of noise enhancement), four bit signal quantization is no longer adequate. Higher resolutions are required to capture these performance gains. From a power perspective, there is a substantial penalty for this increased resolution. A/D power consumption grows exponentially with bit resolution. Power requirements for the digital section also increase because of the additional processing load. Current research focuses on quantifying the power-performance trade-off.

6. Conclusions

Portability imposes severe constraints on wireless system design. Receiver power consumption is often the critical factor affecting design choices. The demands of wireless multimedia access promise to accelerate this trend. Low power design methodologies are needed to address these concerns.

This paper considers one such methodology and its application to the design of a wireless multimedia access system. Using this approach, an integrated CMOS receiver has been implemented which achieves total power consumption of less than 150 mW for a 1 Mbps data rate. These results motivate application of the methodology to more sophisticated architectures, yielding higher performance receivers suitable for portable applications.

References

[1] A. Chandrakasan, S. Sheng, R. Brodersen, "Low Power CMOS Digital Design," *IEEE Journal of Solid-State Circuits*, 27(4), Apr. 1992.

[2] A. Chandrakasan, R. Brodersen, *Low Power Digital CMOS Design*, Boston: Kluwer Academic, 1995.

[3] S. Sheng, A. Chandrakasan, R. Brodersen, "A Portable Multimedia Terminal," *IEEE Comm. Mag.*, Dec. 1992.

[4] S. Sheng, et al, "A Low Power CMOS Chipset for Spread Spectrum Communications," *IEEE Intern. Solid-State Circuits Conf.*, Feb. 1996.

[5] J. Linnartz, L. Yun, M. Couture. "Multi-user and Self Interference Effects in a QPSK DS-CDMA Downlink with a Ricean Dispersive Channel," *Wireless Personal Communications*, vol. 1, pp. 95-102, 1995.

[6] D. Sarwate, and M. Pursley, "Crosscorrelation Properties of Pseudorandom and Related Sequences," *Proc. IEEE*, Vol. 68, No. 5, May 1980.

[7] J. Proakis, Digital Communications. New York: McGraw Hill, 1989.

[8] R. Lupas, S. Verdu, "Linear Multiuser Detectors for Synchronous Code-Division Multiple Access Channels," *IEEE Trans. Inform. Theory*, vol. IT-35, Jan. 1989.

[9] Z. Zvonar, "Combined Multiuser Detection and Diversity Reception for Wireless CDMA Systems," *IEEE Trans. Vehicular Tech.*, Feb. 1996.

[10] Z. Zvonar, D. Brady, "Suboptimal Multiuser Detector for Frequency-Selective Rayleigh Fading Synchronous CDMA Channels," *IEEE Trans. Commun.*, Feb./Apr. 1995.

[11] A. Duel-Hallen, "Decorrelating Decision-Feedback Multiuser Detector for Synchronous Code-Division Multiple-Access Channel," *IEEE Trans. Commun.*, Feb 1993.

[12] P. Patel, J. Holtzman, "Analysis of a Simple Successive Interference Cancellation Scheme in a DS/CDMA System," *IEEE J. Select. Areas Commun.*, June 1994.

[13] S. Vasudevan, M. Varanasi, "Optimum Diversity Combiner Based Multiuser Detection for Time-Dispersive Rician Fading CDMA Channels," *J. Select. Areas Commun.*, May 1994.

[14] S. Verdu, "Computational Complexity of Optimal Multiuser Detection," Algorithmica, vol. 4, pp. 303-312, 1989.

[15] Z. Zvonar, D. Brady, "Linear Multipath-Decorrelating Receivers for CDMA Frequency-Selective Fading Channels," *IEEE Trans. Commun.*, June 1996.

[16] C. Kchao, G. Stuber, "Performance Analysis of a Single Cell Direct Sequence Mobile Radio System," *IEEE Trans. Commun.*, Oct. 1993.

[17] R. Bultitude, S. Mahmoud, W. Sullivan, "A Comparison of Indoor Radio Propagation Characteristics at 910 MHz and 1.75 GHz," *IEEE J. Select. Areas Commun.*, Jan. 1989.

[18] S. Howard, K. Pahlavan, "Measurement and Analysis of the Indoor Radio Channel in the Frequency Domain," *IEEE Trans. Instrumentation and Measurement*, Oct. 1990.

[19] T. Rappaport, S. Seidel, K. Takamizawa, "Statistical Channel Impulse Response Models for Factory and Open Plan Building Radio Communication System Design," *IEEE Trans. Commun.*, May 1991.

[20] H. Hashemi, "Impulse Response Modeling of Indoor Radio Propagation Channels," *IEEE J. Select. Areas Commun.*, Sept. 1993.

Decoding Metrics for the IS-95 Reverse Link

Colin D. Frank
Motorola Corporate Research
Schaumburg, Illinois

Fuyun Ling
Motorola Cellular Infrastructure Group
Arlington Heights, Illinois

Abstract: The IS-95 reverse link modulation chain consists of a rate 1/3 convolutional code, a bit-level interleaver, and a 64-ary orthogonal modulator. The interleaver improves the performance of the convolutional code by introducing time diversity. However, calculation of the log-likelihood ratio for the combination of bit-level interleaving and 64-ary orthogonal modulation is difficult and may require detailed knowledge of the channel fading process. Several decoding metrics have been published in the literature including the single- and dual-max [1,2]. This paper presents two metrics, the *weighted dual-max* and the *coherent dual-max*, which improve on the performance of the dual-max by 0.3 dB and 1.2 dB respectively.

1. Introduction

The IS-95 reverse link modulation chain consists of a rate 1/3 convolutional code, a bit-level interleaver, and a 64-ary orthogonal modulator. The interleaver improves the performance of the convolutional code by introducing time diversity to break up the channel fading process. However, calculation of the log-likelihood ratio for the combination of bit-level interleaving and 64-ary orthogonal modulation is difficult and may require detailed knowledge of the channel fading process. As a result, approximations to the log-likelihood ratio or other sub-optimal metrics must be used with the Viterbi decoder.

Several metrics have been published in the literature including the single-max [1] and the dual-max [2]. This paper presents two metrics which improve on the performance of the dual-max. The first metric, referred to as the *weighted dual-max*, is a noncoherent decoding metric which combines an estimate of the signal-to-noise ratio with the dual-max metric. The second metric, the *coherent dual-max*, is similar to the dual-max metric except that it employs an estimate of the complex channel gain. These metrics improve on the performance of the dual-max by 0.3 dB and 1.2 dB respectively.

The organization of the paper is as follows. In Section 2, a review of the relevant aspects of the IS-95 reverse link modulation chain is presented, along with a description of the link frame structure and power control algorithm. Section 3 presents derivations of the dual-max, weighted dual-max, and coherent dual-max metrics.

2. IS-95 Reverse Link

The modulation chain for the IS-95 reverse link is provided in Figure 1. The voice or data information arrives at the rate 1/3 convolutional encoder at one of four possible rates: 9600 bps, 4800 bps, 2400 bps or 1200 bps. The data is partitioned into 20 msec frames which contain variously 192, 96, 48, and 24 bits, according to the data rate, and are referred to as full-rate, half-rate, quarter-rate, and eighth-rate frames, respectively.

The output of the convolutional encoder is *bit-interleaved* as specified by IS-95. The term bit-interleaved is intended to differentiate this interleaver from an l-ary interleaver, which partitions the binary input sequence into blocks of l-bits that are kept together during the interleaving process. (The application of l-ary interleaving to the IS-95 reverse link was investigated previously in [3] and shown to significantly outperform a reverse link using

bit-level interleaving and the single-max metric). The interleaver output is partitioned into blocks of 6 bits, each of which is used as an index into the set of length 64 binary-valued Walsh codes.

The selection of the IS-95 reverse link modulation scheme seems to have been influenced by the belief that the inclusion of a pilot on each mobile-to-base transmission would be spectrum inefficient (as opposed to the forward link where a common pilot can be shared by all of the mobiles in the sector). For a noncoherent receiver, repetition coding degrades link performance if the energy per information symbol of the coded and uncoded systems is held equal. To avoid this noncoherent combining loss, fractional-rate frames are transmitted by turning the transmitter on and off in a pseudo-random fashion within a frame.

The reverse link frame is divided into 16 *power control groups* that are used both for power control and for bursting the transmitter on and off. A power control group spans an interval of 6 Walsh codes, or equivalently, a time interval of 1.25 msec. For each fractional-rate frame, a corresponding fraction of the power control groups is pseudo-randomly selected and used to transmit the frame data. Thus, eighth-rate, quarter-rate, and half-rate frames select 2, 4, and 8 power control groups, respectively.

The transmit power of the mobile is held constant for all 6 of the Walsh codes in a power control group. The base station receiver measures the average power received during the power control group and orders the mobile transmitter to increase or decrease its power by 1 dB, according to whether the measured power was above or below a threshold value. Since power control groups span 1.25 msec, the maximum rate of power control on the reverse link is 800 Hz. However, during the transmission of eighth-rate frames, the power control rate is only 100 Hz.

3. Decoding Metrics

The channel for the reverse link will in general consist of m paths. Optimal noncoherent detection is employed for each of the m paths. Let $\mathbf{r}^l = (r_1^l, r_2^l, \ldots, r_M^l)$ denote the vector output of the envelope detector for the l-th path. The problem is simplified by assuming that outputs of the m noncoherent detectors are squared and combined with equal weighting. Therefore, let $\mathbf{z} = (z_1, z_2, \ldots, z_M)$ denote the vector of the combined detector outputs so that $z_j = \sum_l (r_j^l)^2$.

Let the vector $\mathbf{b} = (b_1, b_2, \ldots, b_6)$ denote the six binary symbols corresponding to an Walsh symbol. For each binary symbol b_i, let S_0^i denote the indices of the set of Walsh symbols for which $b_i = 0$ and let S_1^i denote the complement. In all cases, the optimal metric for decoding is the log-likelihood ratio. Consider a channel consisting of m independent equal-energy Rayleigh faded paths having equal noise variance N_0. The form of the log-likelihood depends on how much channel information is available to the receiver. If the average energy per path, $\overline{E}_b$, and noise spectral density, N_0, of the additive white Gaussian noise (AWGN) are known to the receiver (but the instantaneous energy per path is not known), the log-likelihood ratio for the i-th binary symbol is given by

$$\log\left(\frac{\mathrm{pr}\left(\mathbf{z}\middle| b_i = 0, \overline{E}_b / N_0\right)}{\mathrm{pr}\left(\mathbf{z}\middle| b_i = 1, \overline{E}_b / N_0\right)}\right) = \log\left(\frac{\sum_{j \in S_0^i} \exp\left[\left(z_j + 2\overline{E}_b / N_0\right) / \left(1 + 2\overline{E}_b / N_0\right)\right]}{\sum_{k \in S_1^i} \exp\left[\left(z_k + 2\overline{E}_b / N_0\right) / \left(1 + 2\overline{E}_b / N_0\right)\right]}\right).$$

If instead the total instantaneous received energy $E_{tot} = \sum E_i$ of the m paths is known to the receiver, the log-likelihood ratio for the i-th binary symbol is given by

$$\log\left(\frac{\mathrm{pr}(\mathbf{z}|b_i=0,E_{\mathrm{tot}}/N_0)}{\mathrm{pr}(\mathbf{z}|b_i=1,E_{\mathrm{tot}}/N_0)}\right)=\log\left(\frac{\sum_{j\in S_0^i} z_j^{(m-1)/2} I_{m-1}\left(2\sqrt{2z_jE_{tot}/N_0}\right)}{\sum_{k\in S_1^i} z_k^{(m-1)/2} I_{m-1}\left(2\sqrt{2z_kE_{tot}/N_0}\right)}\right).$$

where $I_a(x)$ denotes the a-th order modified Bessel function of the first kind. Note that this second log-likelihood ratio applies even if the average energies of the m paths are not equal.

3.a. Single-Max Metric

The metrics above are undesirable due both to their complexity and to their dependence on detailed assumptions about the channel model. The calculation of the log-likelihood ratios given above require the summation of either 64 exponentials or 64 Bessel functions, and the first log-likelihood ratio applies only if the multipath is independent, Rayleigh distributed (*with* power control), and of equal average energy.

Some metrics have been proposed which require less computation than those given above. One such metric was proposed by Viterbi et. al. [1] and is referred to here as the *single-max* metric. Let z_{max} denote the largest element of the vector $\mathbf{z}$. The single-max log-likelihood ratio is given by

$$\log\left(\frac{\mathrm{pr}(z_{\max}=x|b_i=0)}{\mathrm{pr}(z_{\max}=x|b_i=1)}\right)=(-1)^{J(i,z_{\max})}\log\left(1-\frac{1}{32}+\frac{1}{32}\frac{f_{nc}^m(x)F_c^m(x)}{f_c^m(x)F_{nc}^m(x)}\right).$$

where

$$J(i,z_{\max})=\begin{cases}0 & \text{if the index of } z_{\max}\in S_0^i\\ 1 & \text{if the index of } z_{\max}\in S_1^i\end{cases},$$

$f_c^m(x)$ and $F_c^m(x)$ denote the density and distribution function of the noise-only elements of $\mathbf{z}$, and $f_{nc}^m(x)$, and $F_{nc}^m(x)$ denote the density and distribution function of the signal element of $\mathbf{z}$. The noise-only distributions are given by

$$f_c^m(x)=\frac{x^{m-1}e^{-x}}{(m-1)!}\qquad F_c^m(x)=1-e^{-x}\sum_{j=1}^{m}\frac{x^{j-1}}{(j-1)!}.$$

The signal present distributions are given by

$$f_{nc}^m(x)=\frac{x^{m-1}e^{-x/(1+2\bar{E}_b/N_0)}}{(m-1)!\left(1+2\bar{E}_b/N_0\right)^m}\qquad F_c^m(x)=1-e^{-x/(1+2\bar{E}_b/N_0)}\sum_{j=1}^{m}\frac{1}{(j-1)!}\left(\frac{-x}{1+2\bar{E}_b/N_0}\right)^{j-1},$$

for a channel with m independent, equal-energy, Rayleigh fading channels, and by

$$f_{nc}^m(x)=\left(\frac{x}{2E_{tot}/N_0}\right)^{(m-1)/2}\exp^{-(x+2E_{tot}/N_0)}I_0\left(2\sqrt{2xE_{tot}/N_0}\right)$$

$$F_{nc}^m(x)=1-Q_m\left(2\sqrt{E_{tot}/N_0},\sqrt{2y}\right)$$

for the m-path channel with known signal-to-noise E_{tot}/N_0.

Although the single-max log-likelihood ratio is simpler in the sense that it depends only on z_{max} rather than the vector $\mathbf{z}$, the above equations are still dependent on assumptions

regarding the channel model as well as estimates of either $\overline{E}_b/N_0$ or E_{tot}/N_0. However, the single-max log-likelihood ratio can be approximated by

$$\log\left(\frac{\mathrm{pr}(z_{\max}|b_i=0)}{\mathrm{pr}(z_{\max}|b_i=1)}\right) \approx c\cdot(-1)^{y(i,z_{\max})}\, z_{\max},$$

where c is a constant. This last approximation for the log-likelihood ratio is referred to as the *single-max* metric [1].

3.b. Dual-Max Metrics

Let $z^i_{j,\max}$ denote the largest element of the set S^i_j. Refer to the log-likelihood ratios for the vector **z** given above and note that largest terms of the numerator and denominator are functions of $z^i_{0,\max}$ and $z^i_{1,\max}$, respectively. This observation suggests two alternative methods for simplifying the metric computation. One alternative is to approximate both the numerator and the denominator by these largest terms. However, since the resulting approximation depends only on $z^i_{0,\max}$ and $z^i_{1,\max}$, a more direct approach is to evaluate the dual-max log-likelihood ratio given by

$$\log\left(\frac{\mathrm{pr}(z^i_{0,\max}=x,\, z^i_{1,\max}=y|\, b_i=0)}{\mathrm{pr}(z^i_{0,\max}=x,\, z^i_{1,\max}=y|\, b_i=1)}\right) =$$

$$\log\left(\left(\frac{f^m_{nc}(x)F^m_c(x)+15\cdot F^m_{nc}(x)f^m_c(x)}{f^m_{nc}(y)F^m_c(y)+15\cdot F^m_{nc}(y)f^m_c(y)}\right)\left(\frac{f^m_c(y)F^m_c(y)}{f^m_c(x)F^m_c(x)}\right)\right).$$

The dual-max log-likelihood will perform at least as well as the single-max log-likelihood since $z^i_{0,\max}$ and $z^i_{1,\max}$ are sufficient for $z_{\max}$.

Investigation of the dual-max log-likelihood ratio yields some simple approximations. For the m-path Rayleigh fading channel, the log-likelihood ratio can be approximated by

$$\log\left(\frac{\mathrm{pr}(z^i_{0,\max}=x,\, z^i_{1,\max}=y|\, b_i=0,\, \overline{E}_b/N_0)}{\mathrm{pr}(z^i_{0,\max}=x,\, z^i_{1,\max}=y|\, b_i=1,\, \overline{E}_b/N_0)}\right) \approx c\cdot(x-y),$$

where the constant c has no affect on which path is selected by decoder and can therefore be set arbitrarily in the implementation. This approximation of the likelihood ratio is referred to as the *dual-max metric* and was described by Viterbi in [2]. Note that no estimate of $\overline{E}_b/N_0$ is required to implement the metric. If E_{tot}/N_0 is known (or can be estimated) at the receiver, the log-likelihood ratio can be approximated by

$$\log\left(\frac{\mathrm{pr}(z^i_{0,\max}=x,\, z^i_{1,\max}=y|\, b_i=0,\, E_{tot}/N_0)}{\mathrm{pr}(z^i_{0,\max}=x,\, z^i_{1,\max}=y|\, b_i=1,\, E_{tot}/N_0)}\right) \approx c\cdot(E_{tot}/N_0)(x-y),$$

where c is a constant. This second metric will be referred to as the *weighted dual-max metric*.

While the E_{tot}/N_0 will not be known by receiver, this quantity can be estimated from the received data if the channel changes sufficiently slowly. Recall that the uplink data is partitioned into power control groups of six Walsh symbols and that the transmitted power is constant during a power control group. If the channel is relatively static during the span of a power control group, the signal-to-interference ratio can be estimated and applied to all

of the metric computations in that power control group. Also, note that an estimate of the received signal energy is already available at the receiver since IS-95 requires that the received signal energy be measured and used in the power control algorithm. A description of the method used to estimate E_{tot}/N_0 is given in the Appendix.

Figures 2 and 3 illustrate the performance of the weighted dual-max metric, as well as the previous single- and dual-max metrics. These simulation results are for vehicles traveling 30 mph and 60 mph, respectively. The channel is a power controlled 2-path Rayleigh fading channel and the carrier frequency is 900 MHz. Note that the performance of the weighted dual-max metric is approximately 0.3 dB better than the dual-max metric in terms of both the decoded bit error rate and the frame error rate for error probabilities of 10^{-2} and 10^{-3}, respectively.

3.c. Coherent Dual-Max

The metrics generated above are all based on noncoherent demodulation methods. However, recent investigations have shown that effective methods exist for computing good estimates of the complex channel gain on the reverse link. Specifically, an estimate of the complex channel gains for the channel multipath can be obtained by applying the method of near-maximum-likelihood sequence estimation (NMLSE) presented in [4] to each power control group. With these estimates of the complex channel gains, each multipath component can be coherently demodulated. A description of a method of the NMLSE method can be found in [4].

Demodulation is accomplished in the following manner. The NMLSE algorithm yields a near maximum-likelihood estimate of the sequence of 6 Walsh codes received within a given power control group. This sequence of Walsh codes is then correlated with the received signal at delays corresponding to the set of multipath delays. The results of these complex correlations are estimates of the multipath complex channel gains.

Simultaneously with the above, Fast Hadamard Transforms (FHT) are performed on the received signal at delays corresponding to the multipath delays. Each FHT yields a complex vector of length 64, which is multiplied by the complex conjugate of the gain of the corresponding channel multipath. The multipath can be coherently combined by summing the real part of the resulting vectors. The resulting real vector, denoted by $\mathbf{r}$ (r_i, i = 1, ..., 64), can be shown to be a sufficient statistic.

The log-likelihood ratio for the i-th binary symbol given the full vector $\mathbf{r}$ is given by

$$\log\left(\frac{\mathrm{pr}(\mathbf{r}|\, b_i = 0)}{\mathrm{pr}(\mathbf{r}|\, b_i = 1)}\right) = \log\left(\frac{\sum_{j \in S_0^i} \exp(r_j)}{\sum_{k \in S_1^i} \exp(r_k)}\right).$$

As in the noncoherent case, calculation of this log-likelihood ratio requires the calculation of 64 exponentials.

Following the development of the noncoherent dual-max, let $r^i_{j,\max}$ denote the largest element of $\mathbf{r}^i$ with index in the set S^i_j. As above, an expression can be written for the coherent dual-max log-likelihood ratio. Empirical analysis has shown that the coherent dual-max log-likelihood ratio can be well approximated by a square function within the region of significance; that is,

$$\log\left(\frac{\mathrm{pr}(r^i_{0,\max}=x,\, r^i_{1,\max}=y \,|\, b_i=0)}{\mathrm{pr}(r^i_{0,\max}=x,\, r^i_{1,\max}=y \,|\, b_i=1)}\right) \approx c\cdot(x^2-y^2).$$

From Figure 4, it can be show that the coherent dual-max performs approximately 1.2 dB better than the weighted dual-max metric. However, it is important to note that this gain will only be achieved if the coherence time of the channel is longer than 1.25 ms. The minimum coherence time is typically greater than 1.25 msec in cellular or PCS systems, but the method may not be appropriate for systems with very fast fading channels. It should also be noted that the performance of the weighted dual-max metric shown in Figure 4 is slightly worse than shown in Figure 3, because a longer power control delay was used in the simulation.

4. Conclusions

Two improvements to the dual-max decoding metric have been presented which significantly enhance the performance of the IS-95 reverse link at the cost of increased complexity. In particular, we have shown that the coherent dual-max algorithm can improve the sensitivity of the reverse link by up to 1.2 dB. This improved uplink receiver sensitivity will translate directly into increased system capacity and/or increased communication range.

Appendix: A Method for Estimating the Power Control Group Signal-to-Noise Ratio

Let the vector $\mathbf{z}^{(p)}$ denote the vector output of the equal gain combiner for the p -th Walsh symbol within the power control group, $1 \le p \le 6$. We then have the estimates

$$\hat{N}_0^{(p)} = \sum_{\left\{k: z_k^{(p)} \neq z_{\max}^{(p)}\right\}} z_k^{(p)} / 63 \qquad \hat{E}_W^{(p)} = z_{\max}^{(p)} - \hat{N}_0^{(p)},$$

of the noise and signal, respectively, for the p -th Walsh symbol, where $z_{\max}^{(p)} = \max\left\{z_k^{(p)}\right\}$. The estimate $\hat{E}_W / N_0$ for the power control group is then given by

$$\hat{E}_W / N_0 = \left(\sum_{1\le p\le 6} \hat{E}_W^{(p)}\right) \Big/ \left(\sum_{1\le p\le 6} \hat{N}_0^{(p)}\right).$$

Note that the complexity of the noise estimate $\hat{N}_0^{(p)}$ can be reduced by computing the estimate over a subset of $\left\{z_k^{(p)} : z_k^{(p)} \neq z_{\max}^{(p)}\right\}$.

Acknowledgments:

The authors would like to thank Motorola Cellular Infrastructure Group (CIG) and Motorola Corporate Research for supporting this work and allowing its publication. We would also like to thank Mike Schaffner of Motorola CIG for providing the simulation results on the coherent dual-max.

References:

[1] A. J. Viterbi, A. M. Viterbi, and E. Zehavi, "Performance of Power-Controlled Wideband Terrestrial Digital Communication," IEEE Trans. on Comm., vol. 41, No. 4, April, 1993.

[2] A. J. Viterbi, *CDMA: Principles of Spread Spectrum Multiple Access Communication*, Reading, Massachusetts: Addison-Wesley, 1995.

[3] L. F. Chang, D. D. Falconer, F. Ling, and N. Sollenberger, "Comparison of Two Convolutional Coding Techniques for CDMA Radio Communication Systems," IEEE Trans. on Comm., Vol. 43, No. 6, pp. 2028-2037, June, 1995.

[4] R. Walton and M. Wallace, "Near Maximum Likelihood Demodulation for M-ary Orthogonal Signaling," Proceedings of the 1993 Vehicular Technology Conference, pp. 5-8.

[5] Y. Liu, "Soft Decision Decoding of a Bit-Interleaved Convolutionally Encoded Code Division Multiple Access System over Rayleigh Fading Channels," 1992 Personal, Indoor, and Mobile Radio Communications Conference, pp. 4.7.1-5.

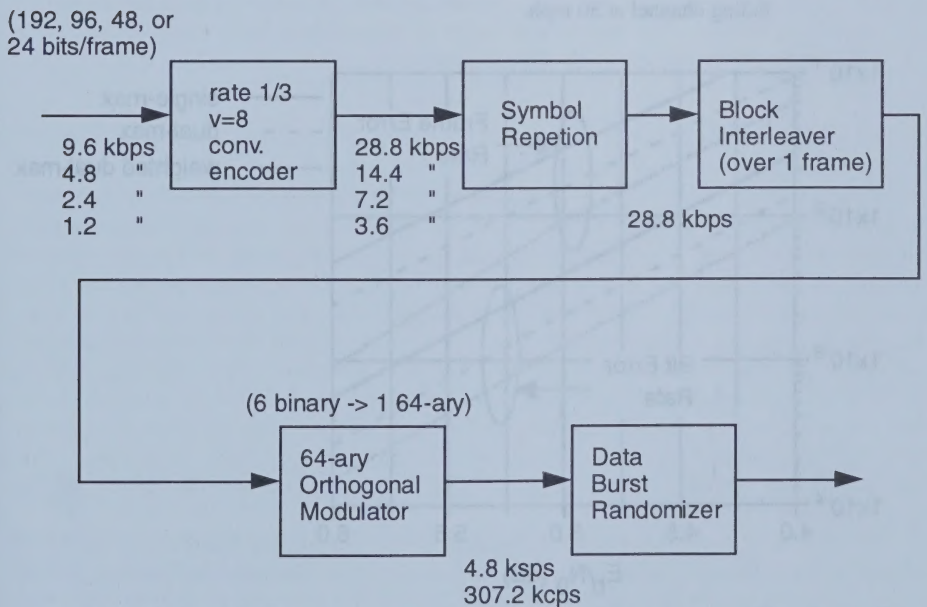


Figure 1: IS-95 Reverse Link Modulation and Coding

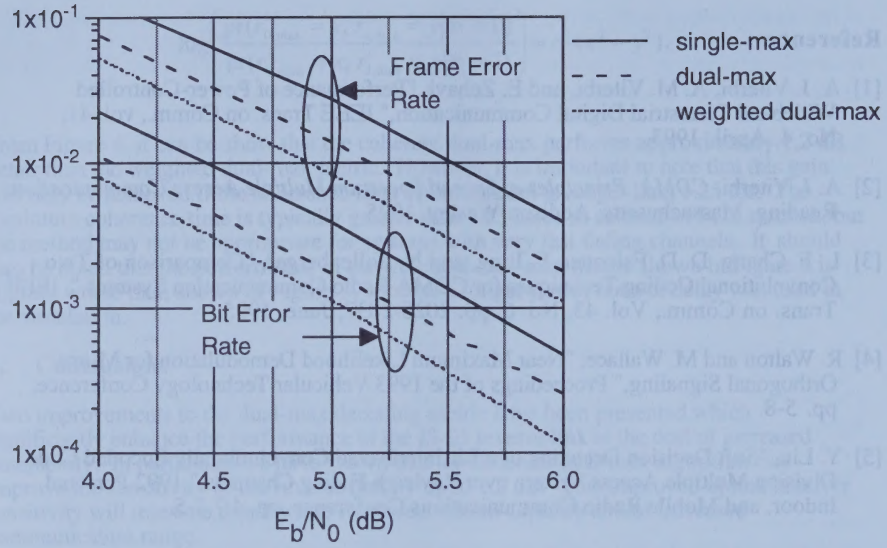


Figure 2. Bit error rate and frame error rate for a power controlled 2-path Rayleigh fading channel at 30 mph.

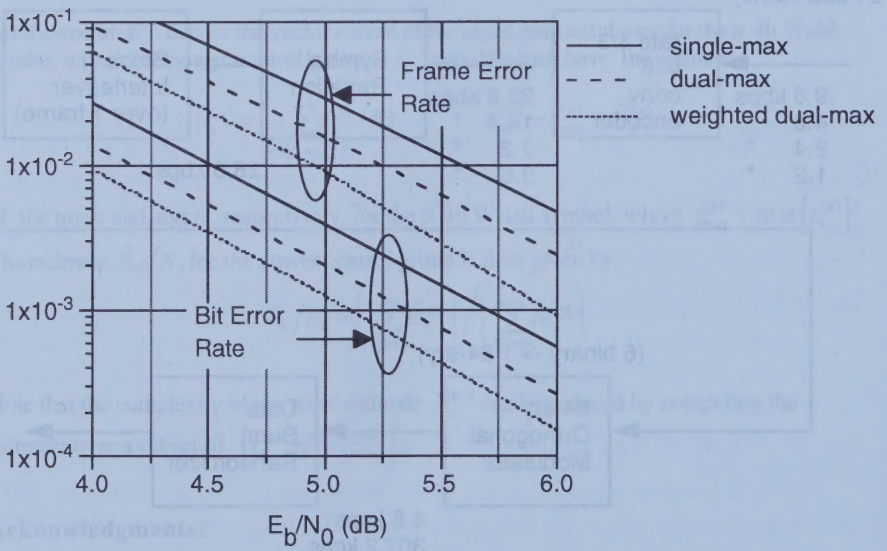


Figure 3. Bit error rate and frame error rate for a power controlled 2-path Rayleigh fading channel at 60 mph.

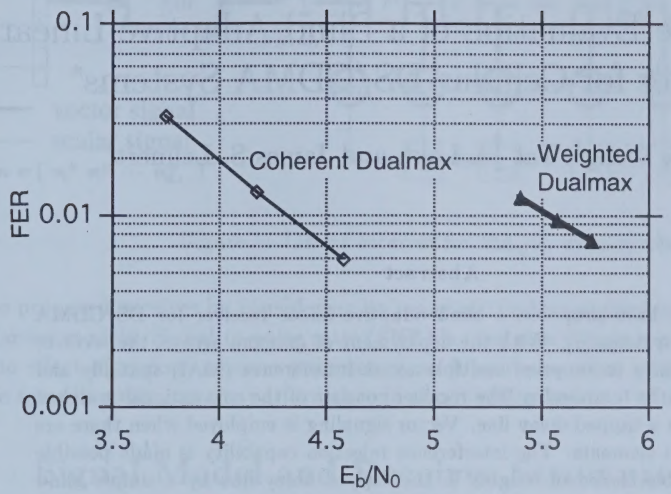


Figure 4. Frame error rate with the coherent dual-max metric for a power controlled 2-path Rayleigh fading channel at 60 mph.

Performance Evaluation of a Blind Adaptive Linear Receiver for Cellular DS/CDMA Systems*

Tan F. Wong†, Tat M. Lok‡, and James S. Lehnert†

Abstract

Recently, we have proposed a blind adaptive linear receiver for DS/CDMA communication systems [1]. With the deployment of antenna arrays, the receiver has the joint ability to suppress multiple-access interference (MAI) spatially and combine multipaths temporally. The receiver consists of the conventional matched filter followed by a tapped delay line. Vector signaling is employed when there are multiple antenna elements. The interference rejection capability is made possible through a suitable choice of weights in the tapped delay line by a simple blind adaptive algorithm. The simplicity and the capability for adaptive interference suppression by the receiver make it suitable for cellular wireless applications. In this paper via computer simulations, we evaluate the performance of such a receiver under cellular multiple-access communication models with multipath fading and spatial variations. The goal of this paper is to provide a further step toward practical implementations of the proposed receiver.

1 Introduction

It is well-known [4–6] that the capacity of a cellular CDMA system can be increased by the use of antenna arrays. One of the challenges, however, is to develop a signaling method together with a receiving structure which can effectively utilize the antenna arrays to achieve capacity improvement. Moreover, such a receiving structure has to be reasonably simple in order to be implemented practically. Aiming to achieve this goal, we have proposed a blind adaptive linear receiver employing direct-sequence spread-spectrum signaling techniques for the use of antenna arrays in cellular CDMA systems [1]. The receiver consists of the conventional matched filter followed by a tapped delay line. Vector signaling is employed when there are multiple antenna elements. In addition to providing spatial selectivity, the receiver can combine multipaths temporally. This makes it effective in combating frequency-selective fading which often occurs in cellular environments.

We describe the receiver structure together with the signaling method and the blind adaptive algorithm in Section 2 and 3. In Section 4, we evaluate the performance of

*This material is based upon work supported by the U. S. Army Research Office under grant number DAAH04-95-1-0246.

†T. F. Wong and J. S. Lehnert are with the School of Electrical and Computer Engineering, Purdue University, West Lafayette, IN 47907-1285.

‡T. M. Lok is with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong.

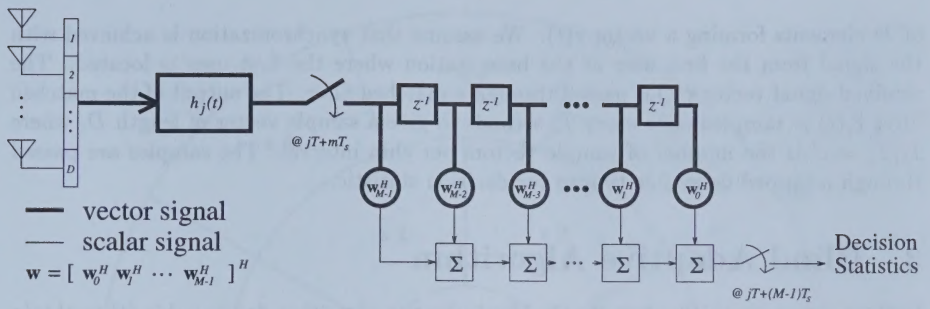


Figure 1: Linear receiver for the jth data symbol

the proposed receiver by considering its usage at the base station under two reverse link channel models. Signal-to-noise ratio (SNR) is used as a simple measure of performance. The effect of various receiver parameters, such as the number of antenna elements and the sampling rate, on the performance is also investigated.

2 Signal Model and Receiver Structure

In this section, we describe the format of the signals generated by the mobile users and the structure of the receivers at the base stations. We consider a special case of the signal format detailed in [1]. The kth user in the system generates a data signal $b_k(t)$ that is given by

$$b_k(t) = \sum_{j=-\infty}^{\infty} b_j^{(k)} p_T(t - jT), \tag{1}$$

where T is the symbol duration, and $p_T(t)$ is the unit rectangular pulse of duration T. The data symbols $b_j^{(k)}$ are independent Bernoulli random variables taking values from $\{+1, -1\}$ with equal probability (i.e., BPSK signaling). The kth user is provided a pseudo-random signature sequence $a^{(k)} = (\ldots, a_0^{(k)}, a_1^{(k)}, \ldots, a_{N-1}^{(k)}, \ldots)$, which is a shifted version of the $(2^{42} - 1)$-length binary sequence given in IS-95 [2]. We note that different users are assigned different shifts of the $(2^{42} - 1)$-length sequence. Hence, the correlation properties of this set of signature sequences are close to those of the random sequence model assumed in [1]. The sequence $a^{(k)}$ is used to form the spectral spreading signal that is given by

$$a_k(t) = \sum_{i=-\infty}^{\infty} a_i^{(k)} \psi\Big(\frac{t - iT_c}{T_c}\Big), \tag{2}$$

where the chip duration T_c of the common rectangular chip waveform is given by $T_c = T/N$, N is the number of chips per symbol interval, and $\psi(t) = 1$ for $0 \leq t < 1$, and $\psi(t) = 0$, otherwise. The transmitted signal for the kth user can be expressed as $\mathrm{Re}[z_k(t - T_k)\exp(\mathrm{j}\omega_c(t - T_k))]$, where $z_k(t) = \sqrt{2P_{t,k}}a_k(t)b_k(t)$, $P_{t,k}$ is the transmitted power for the kth signal, ω_c is the carrier frequency, and T_k is the delay that models the asynchronous system. Without loss of generality, we will consider the signal from the first user as the communicating signal and the signals from all other users as interfering signals throughout the paper.

The structure of the receiver is shown in Fig. 1 and is detailed in [1]. We give a brief review here. The signals are received at each base station by an antenna array

of D elements forming a vector $\mathbf{r}(t)$. We assume that synchronization is achieved with the signal from the first user at the base station where the first user is located. The received signal vector $\mathbf{r}(t)$ is passed through a matched filter. The output of the matched filter $\tilde{\mathbf{r}}_j(t)$ is sampled once every T_s seconds to give a sample vector of length D, where $T_c/T_s = S$ is the number of sample vectors per chip interval. The samples are passed through a tapped delay line to give the decision statistic.

3 Blind Adaptive Algorithm

In this section, we briefly describe the blind adaptive algorithm developed in [1] to obtain the optimal weights in the tapped delay line in order to achieve spatial and temporal selectivity.

For the detection of the jth symbol, we sample the output of the matched filter in Fig. 1 every T_s seconds and get M sample vectors so that all the paths, corresponding to the jth symbol, of the first user are included in the sampling period of MT_s seconds. We, then, stack the M sample vectors together to form the vector $\mathbf{s}_j$ of length MD. After some chip durations, we sample the output of the matched filter again. This time we obtain another M sample vectors so that no path, corresponding to the jth symbol, of the first user is included in this sampling period. Again, we stack the M sample vectors to get the vector $\hat{\mathbf{s}}_j$ of length MD. Note that the timing difference between each element of $\hat{\mathbf{s}}_j$ and the corresponding element of $\mathbf{s}_j$ should be an integral multiple of the chip duration.

With $\mathbf{s}_j$ and $\hat{\mathbf{s}}_j$, we adjust the weight vector $\mathbf{w}(j)$ for each iteration according to the following rule:

Given any $\mathbf{w}(0)$ *such that* $\mathbf{z}^H\mathbf{w}(0) \neq 0$. *For* $j > 0$, *get* $\mathbf{w}(j)$ *by*

$$\mathbf{w}(j) = c(j)\Big\{\mathbf{w}(j-1) + \mu[\mathbf{s}_j^H\mathbf{w}(j-1)][(\hat{\mathbf{s}}_j^H\hat{\mathbf{s}}_j)\mathbf{s}_j - (\hat{\mathbf{s}}_j^H\mathbf{s}_j)\hat{\mathbf{s}}_j)]\Big\},$$

where $\mathbf{z} = E[b_j^{(1)}\mathbf{s}_j]$ *and* $c(j)$ *is chosen to stabilize the algorithm.*

Note that for the algorithms stated above to give the desired weight vector, we have to initialize $\mathbf{w}(0)$ such that it is not orthogonal to the signal vector $\mathbf{z}$. In practical situations with synchronization achieved, the first sample at the output of the matched filter usually carries a significant portion of power. In most cases, it should not be zero. Hence, it is usually a safe choice to start the algorithms with $\mathbf{w}(0) = [1\ 0\ \ldots\ 0]^T$. By the same line of reasoning, we can choose $c(j)$ as below:

For each $j > 0$, *let* $w_1(j)$ *be the first element of* $\mathbf{w}(j)$. *If* $w_1(j) \neq 0$, *choose* $c(j)$ *such that* $\|\mathbf{w}(j)\| = 1$ *and* $\arg(w_1(j)) = 0$. *Otherwise, choose* $c(j)$ *such that* $\|\mathbf{w}(j)\| = 1$ *and* $\arg(c(j)) = 0$.

The blind adaptive algorithm described above is a constrained stochastic gradient algorithm. Its behavior is expected to be similar to that of the LMS algorithm. As a rule of thumb, we should choose μ such that

$$0 < \mu < \mu_{max} = \frac{2}{(\text{average energy in } \mathbf{s}_j)^2}. \tag{3}$$

We note that the computational complexity of the algorithm is of the order of MD per iteration.

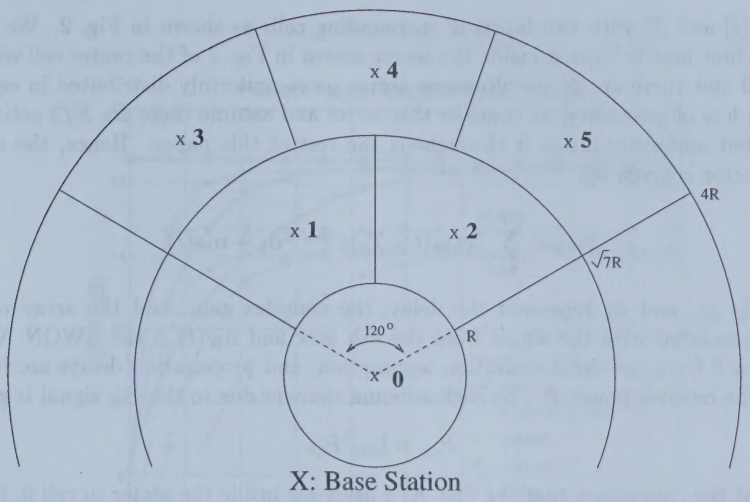


Figure 2: Wedge cell model

4 Performance Evaluation

We need a performance measure to evaluate the performance of the proposed receiver employing the adaptive algorithm described in previous sections. In this paper, we employ the signal-to-noise ratio (SNR) of the decision statistic at the output of the tapped delay line as the performance measure. Being directly related to the bit error probability, the SNR can be easily obtained through computer simulations. For voice communication, it is suggested [3] that a raw bit error rate of 10^{-3} is sufficient when error control coding schemes are employed. Loosely speaking, using Gaussian approximation, this corresponds to the requirement that the SNR is at least 10dB. Due to the statistical nature of the received signal, we examine the average SNR and assume that the performance of a particular system is acceptable when the 10dB limit is one standard deviation below the average SNR.

We evaluate the performance of the proposed receiver under two reverse link channel models. These two models are by no mean adequate in representing the complex cellular environments we encountered. Nevertheless, they possess the essence of a typical cellular environment and, at the same time, are simple enough for analysis purposes. Two different models are considered so that the performance of the proposed receiver in combating different channel defects can be singled out. In both cases, we assume 120° sectorization at the base station. In each sector, there is a linear antenna array with D isotropic elements, which are separated by half a wavelength. Moreover, we assume that the channel is also corrupted by an AWGN with two-sided power spectral density $N_0/2$ and the AWGN components picked up by different antenna elements are independent. We also set $N = 127$ for all the following cases.

4.1 Multiple cell, non-fading model

The first channel model we considered is to demonstrate the spatial selectivity of the proposed receiver. Therefore, no fading is considered. We adopt the wedge cell model

used in [4] and [5] with two layers of surrounding cells as shown in Fig. 2. We assume that the first user is located inside the sector shown in Fig. 2 of the center cell with base station 0 and there are K simultaneous active users uniformly distributed in each cell. Without loss of generality, we consider this sector and assume there are $K/3$ active users distributed uniformly inside it throughout the rest of this paper. Hence, the received signal vector is given by,

$$\mathbf{r}(t) = \sum_{k=1}^{16K/3} g_k z_k(t - T_k) e^{-\mathrm{j}\omega_c T_k} \mathbf{d}_k + \mathbf{n}_W(t), \tag{4}$$

where T_k, g_k, and $\mathbf{d}_k$ represent the delay, the complex gain, and the array response vector associated with the signal from the kth user and $\mathbf{n}_W(t)$ is the AWGN. We note that $T_1 = 0$ from the synchronization assumption, and propagation delays are included in T_k. The receiver power $P_{r,k}$ by each antenna element due to the kth signal is given by

$$P_{r,k} = |g_k|^2 P_{t,k}. \tag{5}$$

We adopt the convention that the first $K/3$ users are inside the sector in cell 0, the next K users are in cell 1, and so on. From (4), since there is only one received path for the desired user, it suffices to take $M = 1$ with the assumption that synchronization is achieved.

We assume that a certain power control scheme is applied and consider the following two cases.

4.1.1 Perfect power control

In the first case, perfect power control is achieved. This means that the mobile users adjust their transmission powers so that the base station of a cell receives signals of the same power, P_0, from all users located inside the cell. We point out that the received power by each antenna element, but not the total received power, is being fixed to P_0. Under this assumption, the transmitted power of a particular mobile user is independent of the number of elements used in the base station antenna. It can be easily shown [5] that

$$P_{r,k} = \begin{cases} P_0 & \text{if } 1 \le k \le K/3, \\ P_0(d_{k,i}/d_{k,0})^n & \text{if } (i-1)K + K/3 + 1 \le k \le iK + K/3, \quad 1 \le i \le 5, \end{cases} \tag{6}$$

where $d_{k,i}$ is the distance of the kth user from the base station i, $0 \le i \le 5$ and n is the path loss exponent. In the rest of this paper, we assume $n = 4$ and $10\log_{10}(P_0 T/N_0) = 9\text{dB}$.

Fig. 3 shows the convergence behavior of the blind adaptive algorithm for a system with $D = 6$ and $K = 30$. We can see from Fig. 3 that the rate of convergence increases as μ increases. For the three values of μ chosen, we can get to within about 0.2dB of the optimal mean SNR. The variances of the SNR's for the three cases are approximately the same. We can get to the 10dB performance requirement within 10 symbol durations by choosing $\mu \le 0.5\mu_{max}$. In the following, we set $\mu = 0.5\mu_{max}$.

Fig. 4 shows the SNR achieved by the blind adaptive algorithm for different values of K and D. By using a single element antenna, 10 active users can be accommodated into a single cell. This number increases to 42 when an antenna array of 3 elements is used. When an array of 6 elements is used, the cell can accommodate more than 66 active users. This accounts for more than a 6-fold increase in capacity.

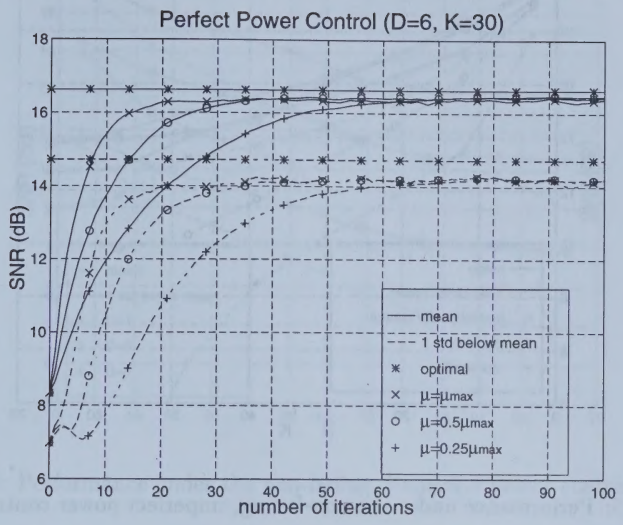


Figure 3: Convergence behavior

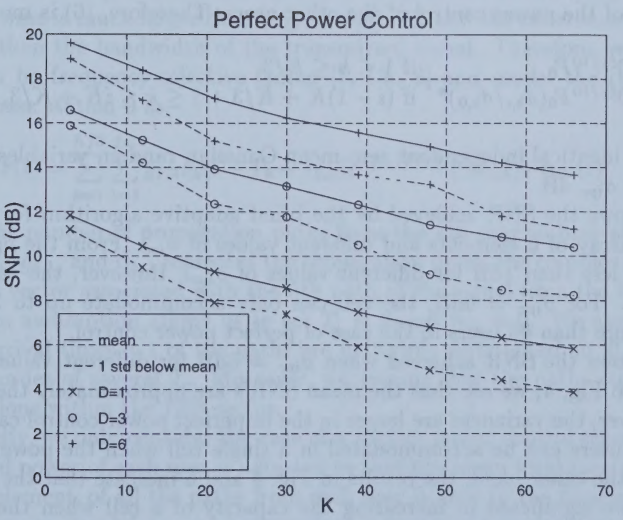


Figure 4: Performance under the non-fading, perfect power control model

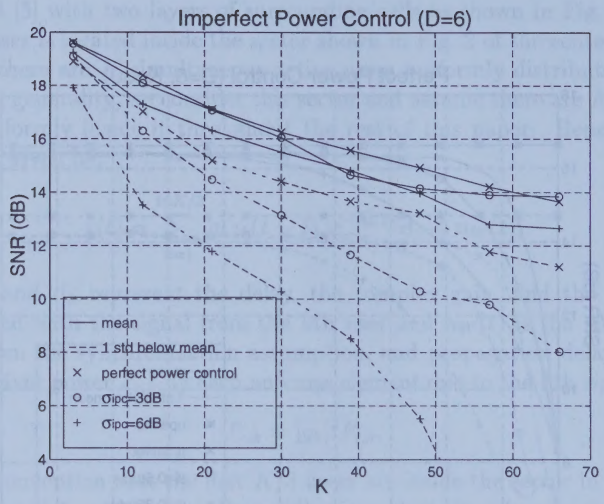


Figure 5: Performance under the non-fading, imperfect power control model

4.1.2 Imperfect power control

Practically, it is impossible to achieve perfect power control. In order to model imperfections in any practical power control scheme, we assume the actual received signal power of an in-cell user at a base station is given by the nominal power P_0 multiplied by a lognormal random variable and the imperfection in the power control of each user is independent of the power control of the other users. Therefore, (6) is modified to

$$P_{r,k} = \begin{cases} 10^{\xi_k/10} P_0 & \text{if } 1 \le k \le K/3, \\ 10^{\xi_k/10} P_0 (d_{k,i}/d_{k,0})^n & \text{if } (i-1)K + K/3 + 1 \le k \le iK + K/3, \quad 1 \le i \le 5, \end{cases} \tag{7}$$

where ξ_k are identical independent zero-mean Gaussian random variables with standard deviations of σ_{ipc} dB.

Fig. 5 shows the SNR achieved by the blind adaptive algorithm for a system with an antenna array of 6 elements and different values of σ_{ipc}. From the figure, the mean SNR's differ less than 1dB for different values of σ_{ipc}. However, the variances change dramatically. For σ_{ipc} = 6dB, the cell can only accommodate up to 30 active users instead of more than 66 users in the case of perfect power control.

Fig. 6 shows the SNR achieved when σ_{ipc} = 3dB for different values of K and D. Comparing to Fig. 4, we see that the mean SNR's are approximately the same for both cases. However, the variances are larger in the imperfect power control case. As a result, fewer active users can be accommodated in a single cell when the power control is not perfect. On the other hand, the results in Fig. 4 and 6 indicate that the use of antenna arrays is more significant in increasing the capacity of a cell when the power control scheme is imperfect. For example, when an array of 3 elements is used, the increase in capacity is about 4-fold for the perfect power control case whereas the increase is 5-fold for σ_{ipc} = 3dB.

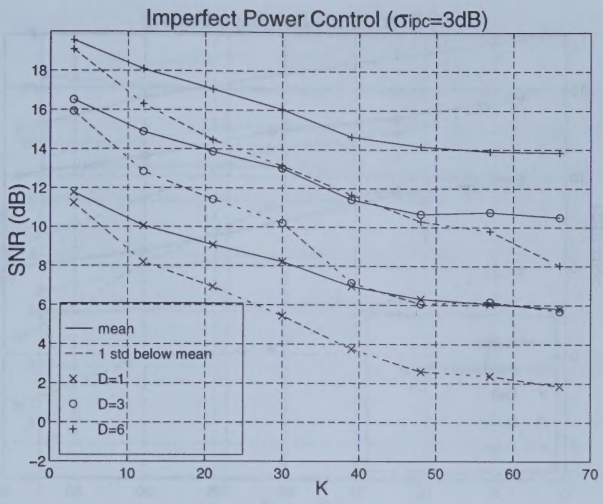


Figure 6: Performance under the non-fading, imperfect power control model

4.2 Single cell, multipath fading model

Now, we develop a model to describe a fading environment. For simplicity, we only consider users in the sector of the center cell shown in Fig. 2 and neglect the effect of the signals transmitted by the users in the outer layer cells. Again, we assume there are $K/3$ active users distributed uniformly in the sector shown in cell 0. For existing practical DS/CDMA systems designed for urban environments, such as the one proposed in IS-95, the coherence time is much larger than chip duration T_c and the coherence bandwidth is much smaller than the bandwidth of the transmitted signal. Therefore, we can assume the channel to be frequency-selective fading [7, Ch. 8] and model the received signal vector at the base station 0 as

$$\mathbf{r}(t) = \sum_{k=1}^{K/3} \sum_{\lambda=1}^{L_k} g_{k,\lambda} z_k(t - T_k - \tau_{k,\lambda}) e^{-\mathrm{j}\omega_c(T_k+\tau_{k,\lambda})} \mathbf{d}_{k,\lambda} + \mathbf{n}_W(t). \tag{8}$$

In (8), L_k is the number of propagation paths from the kth user to base station 0. The parameters $\tau_{k,\lambda}$, $g_{k,\lambda}$, and $\mathbf{d}_{k,\lambda}$ represent the propagation delay, the complex gain, and the array response vector associated with the λth path of the signal from the kth user. The synchronization assumption allows us to set $T_1 = 0$ and $\tau_{1,1} = 0$. We assume that the independent arrival times $\tau_{k,\lambda}$ of the paths are uniformly distributed on $[0, \tau_{max})$, where τ_{max} is in the order of several T_c. Moreover, we assume that the paths come uniformly from all directions within the 120° sector.

For simplicity, we only consider the case of a perfect power control scheme, by which the transmitted power of each user is adjusted so that the mean total received power, by each antenna element, of all the paths from each user is kept to the constant P_0. Under this assumption, we model $g_{k,\lambda}$ as independent zero-mean complex Gaussian random variables such that, for $1 \le k \le K/3$,

$$E\left[\sum_{\lambda=1}^{L_k} |g_{k,\lambda}|^2 P_{t,k}\right] = P_0. \tag{9}$$

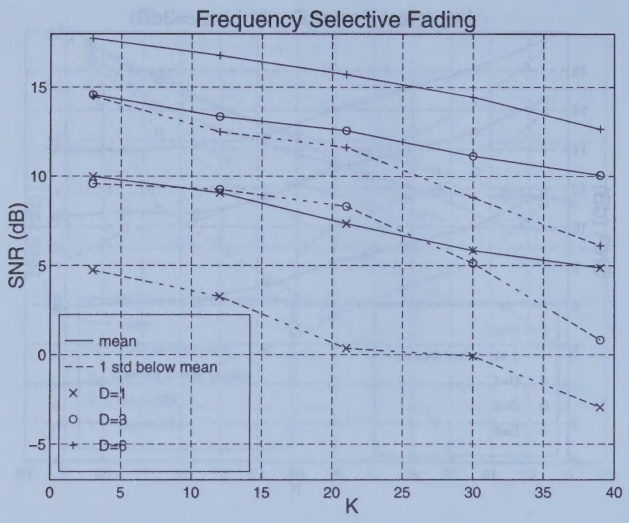


Figure 7: Performance under the multipath fading model

We note that it is implicitly assumed in (9) that all paths of a user have the same path loss. In the following, we take $L_k = 5$, $\tau_{max} = 5T_c$, and thus, $M = 5S$.

Fig. 7 shows the SNR achieved for a system with $S = 1$ and different values of K and D. We see that for $D \leq 3$, the system does not meet the 10dB performance requirement. When $D = 6$, the number of active users a single cell can accommodate is 25, which is much fewer than the non-fading case. We note that this number is optimistic because the effect of the interferers from the outer layer cells are neglected. Therefore, to achieve the same capacity of the non-fading system, we need to increase either D or P_0.

Fig. 8 compares the mean SNR obtained by the blind adaptive algorithm, and the optimal mean SNR can be obtained for different values of S and D. It is as expected that the optimal SNR increases as S increases. However, expect for the case of $D = 1$, the SNR obtained by the algorithm decreases with S due to its LMS-like behavior. In order to get closer to the optimal SNR performance, we have to decrease the value of μ at the expense of reducing the convergence rate of the algorithm.

5 Conclusion

In this paper, we have demonstrated the usefulness of the blind adaptive linear receiver proposed in [1] in increasing the capacity of two reverse link models of cellular systems. The use of the blind adaptive linear receiver in conjunction with antenna arrays is effective for both the fading and non-fading channel models with a certain degree of power control. In fact, the receiver is more effective in increasing the capacity of the non-fading channel model with imperfect power control than that of the one with perfect power control. On the other hand, a drawback of the receiver is that the convergence rate of the weights in the tapped delay line has to be lowered when the number of the weights is large. Otherwise, the SNR performance achieved by the blind adaptive algorithm will not be close to the optimal one.

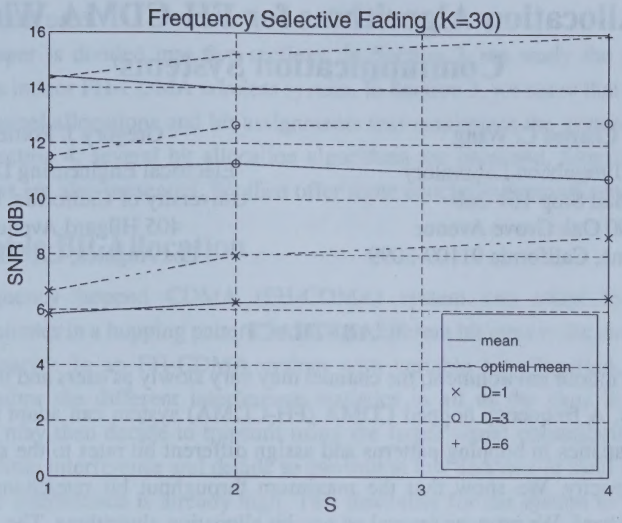


Figure 8: Effect of oversampling

References

[1] T. F. Wong, T. M. Lok, J. S. Lehnert, and M. D. Zoltowski, "A Linear Receiver for Direct-Sequence Spread-Spectrum Multiple-Access Systems with Antenna Arrays and Blind Adaptation," submitted for publication.

[2] TIA/EIA/IS-95 Interim Standard, *Mobile Station-Base Station Compatibility Standard for Dual Mode Wideband Spread Spectrum Cellular System*, Telecommunications Industry Association, Washington, D.C., Jul. 1993.

[3] K. S. Gilhousen, I. M. Jacobs, R. P. Padovani, A. J. Viterbi, L. A. Weaver, and C. E. Wheatley, "On the Capacity of a Cellular CDMA System," *IEEE Trans. Veh. Technol.*, vol. 40, no. 2, pp. 303–312, May 1991.

[4] T. S. Rappaport and L. B. Milstein, "Effects of Radio Propagation Path Loss on DS-CDMA Cellular Frequency Reuse Efficiency for the Reverse Channel," *IEEE Trans. Veh. Technol.*, vol. 41, no. 3, pp. 231–242, Aug. 1992.

[5] J. C. Liberti and T. S. Rappaport, "Analytical Results for Capacity Improvements in CDMA," *IEEE Trans. Veh. Technol.*, vol. 43, no. 3, pp. 680–690, Aug. 1994.

[6] A. F. Naguib, A. Paulraj, and T. Kailath, "Capacity Improvement with Base-Station Antenna Arrays in Cellular CDMA," *IEEE Trans. Veh. Technol.*, vol. 43, no. 3, pp. 691–698, Aug. 1994.

[7] R. L. Peterson, R. E. Ziemer, and D. E. Borth, *Introduction to Spread Spectrum Communications*, Prentice Hall, 1995.

Bit Allocation Algorithms for FH-CDMA Wireless Communication Systems[1]

Charles C. Wang
Jet Propulsion Laboratory
Mail Stop 161-260
4800 Oak Grove Avenue
Pasadena, California 91109-8099

Gregory J. Pottie
Electrical Engineering Department
University of California, Los Angeles
405 Hilgard Avenue
Los Angeles, CA 90024

ABSTRACT

In the wireless indoor environment, the channel may vary slowly as users and interferers move at slow speeds. A frequency hopped CDMA (FH-CDMA) system can adapt to the different interference statistics in hopping patterns and assign different bit rates to the slots to increase the system capacity. We show that the maximum throughput bit rate/channel assignment problem is NP-hard. We propose several ad hoc bit allocation algorithms. The algorithms that achieve the highest system capacity perform interference avoidance. Users concentrate their bits in a small fraction of the slots by transmitting large signal constellations in the slots with low interference while avoiding the channels with large interference. Simulations show that the flexibility of users to adjust their throughputs to the interference environment increases the system capacity significantly.

1 Introduction

Like most of the high capacity wireless systems where the available bandwidth is limited, a slow FH-CDMA system is "interference limited." The transmitted power of every user is seen as unintended signal or interference by all other receivers. Channel assignment strategies that reduce the interference each user generates can directly increase the number of users that can communicate simultaneously. In an indoor wireless FH-CDMA system, the channels may vary slowly as users and interferers move at low speeds. The system can take advantage of the different interference statistics in a hopping pattern and assign different bit rates to the slots to minimize the system interference. Assuming all users desire the same data rate, our goal is to design bit allocation algorithms that can provide services to the most users.

We formulate the problem of allocating bits in FH-CDMA systems in the context of the reverse link (users transmitting to base stations) of a cellular system. The bit allocation algorithms can also be applied to the forward link of a cellular system or a peer-to-peer wireless system. All of the variable bit algorithms described in this paper can be performed in a distributed fashion. No centralized control is necessary. The simulation results presented here are for comparisons of different algorithms and are meant to be upperbounds for the simplified

1. This work is funded by Motorola Inc., under the terms of a university partnerships in research grant.

propagation model used.

This paper is divided into five sections. In Section 2, we study the problem of bit allocation in an indoor FH-CDMA wireless system. In Section 3, we show that the problem of finding the channel allocations and bit assignments that maximizes the system throughput is NP-hard. In Section 4, several bit allocation algorithms are proposed. Simulation results of these algorithms are also presented. We then offer some concluding remarks in Section 5.

2 Variable Bit Allocation

A frequency hopped CDMA (FH-CDMA) system can adapt to the different interference statistics in a hopping pattern and assign different bit rates to the slots to maximize the system capacity. In an FH-CDMA system with variable bit allocation, the users can constantly monitor the different interference statistics in all of the slots in their hopping patterns. They may then decide to transmit using the larger signal constellations in the slots where there is little interference and decide to transmit at low data rate or send no data at all in the slots where interference is already high. This flexibility for the system to assign the data rates and power of each user to accommodate the propagation condition allows the FH-CDMA system to achieve higher system capacity.

At first glance, an FH-CDMA wireless system with variable bit allocation is similar to a multitone wireline communication system [1][2]. In both of these systems, the users first probe the channels that are available to obtain channel statistics. There are several distributed channel probing algorithms proposed for wireless communication systems [3][4][5][6]. A new user would transmit at several different power levels and monitor how the users sharing the same channels react to the interference the new user generates. The maximum achievable signal-to-interference ratio (SIR), the transmitted power needed to achieve the desired SIR, and other statistics regarding power and interference can be estimated. The users in the multitone wireline system and the FH-CDMA system can then allocate bits and power in the slots that have favorable channel conditions. In the multitone wireline communication system, however, the system has an average power constraint. There is a limitation on the total transmitted power permitted for each user. In an FH-CDMA wireless system, there is usually no such limitation, although low transmitted power is desirable as it generally leads to low interference. A maximum transmitted power for every slot due to hardware limitations is more often the case. In addition, an FH-CDMA wireless system is almost always a multi-user system where the transmitted power of one user interferes with the signals of others.

Before we continue our discussion on finding the bit allocation algorithms that maximize the capacity of an FH-CDMA system, we need to briefly discuss the problem of power control. We assume there are B users in the system. In order to maintain the desired error performance, different SIR's are needed for different bit rates/signal constellation sizes. For now, we assume that the users know the bit rates in all slots of the hopping patterns and the SIR needed. If user i needs to obtain SIR greater than or equal to some SIR threshold γ_{ik} to achieve the bit rate that it desires in the k-th slot, its transmitted power needs to satisfy the following inequality:

$$\frac{P_{ik}G_{ii}^{k}}{\sum_{j\neq i}P_{jk}G_{ij}^{k}+n_{ik}} \geq \gamma_{ik} \tag{1}$$

where P_{jk} is the transmitted power of the j-th user in the k-th slot, n_{ik} is equal to the noise floor at the base station communicating with user i, and G_{ij}^{k} is the link gain from user j to the base station communicating with user i. The left hand side of (1) is simply the received power of user i divided by the sum of all of the interference and noise at the receiving base station. We assume that G_{ij}^{k} and n_{ik} are independent of the frequency slots used and omit the superscript k in our notation. We then multiply the denominator of (1) on both side and move all of the terms with the power terms to the left side of the inequality:

$$P_{ik}G_{ii} - \gamma_i \sum_{j\neq i} P_{jk}G_{ij} \geq \gamma_{ik}n_i \tag{2}$$

There are B such linear inequalities, one for each of the users sharing the same slot and B unknowns: $P_{1k}, P_{2k}, \ldots, P_{Bk}$. Dividing G_{ii} for each inequality, we can rewrite the B inequalities in matrix form,

$$H(\vec{\gamma}_k)\,P_k \geq N(\vec{\gamma}_k) \tag{3}$$

$\vec{\gamma}_k$ is a $B \times 1$ vector with the i-th element equals to γ_{ik}. $\vec{\gamma}_k$ is the vector form of the B desired SIR's of the users in the k-th slot. $H(\vec{\gamma}_k)$ is a $B \times B$ matrix. The diagonal elements of $H(\vec{\gamma}_k)$ are equal to one and the off diagonal element (i, j) is equal to $-\gamma_{ik}\,G_{ij} / G_{ii}$. $P_k = [\,p_{ik}\,]$ is the $B \times 1$ non-negative power vector for the set of desired SIR $\vec{\gamma}_k$ in slot k. $N(\vec{\gamma}_k)$ is a $B \times 1$ noise vector where i-th element is equal to $\gamma_{ik}\,n_i / G_{ii}$.

If there exists a non-negative power vector solution to (3), the B users can all achieve the SIR needed for their desired throughputs at the k-th time-frequency slot. A set of linear equations needs to be solved for each time-frequency slot. This set of $B \times B$ linear equations can be solved with power control algorithms with centralized control [7] or without centralized control in a distributed fashion[8][9]. We assume all hopping patterns are periodic with the same period, S slots. The problem of variable bit allocation can be stated as finding the distributions of the SIR thresholds, hence bit allocations, $\{\gamma_{i1}, \gamma_{i2}, \ldots, \gamma_{iS}\}$ for each user i such that the sum of the bit rates of each user is equal to the desired total throughput and that there exist feasible non-negative power solutions in all of S the slots.

3 Complexity of Optimal Bit Allocation Algorithms

We now turn our attention to finding the channel and bit allocations for a given set of users that maximize the system capacity assuming all users must have the same throughput. We will show that the complexity finding the optimal allocation is NP-hard. The optimization problem can be stated in the following compact matrix form:

$$maximize \sum_{i=1}^{B} X_i \tag{4}$$

$$X_i \le \sum_{k=1}^{S} log_2(1+\gamma_{ik}) \tag{5}$$

$$X_i = X_j \qquad 1 \le i, j \le B$$

$$H(\vec{\gamma}_k) P_k \ge N(\vec{\gamma}_k) \tag{6}$$

$$P_k \ge 0 \qquad \vec{\gamma}_k \ge 0 \qquad 1 \le k \le S$$

We try to maximize the total throughput of the system with B users and S slots. X_i is the throughput of the i-th user. The logarithm function with base two (5) is used to upper bound the achievable bit rate as a function of the desired SIR of γ_{ik} [10]. All users have the same throughput. (6) is the set of power constraints described in the last section.

If the number of slots is greater than or equal to the number of users ($B \le S$), the throughput of each user is infinite since we do not impose a maximum power constraint. The infinite throughput is achieved by assigning each user to a different slot. Since there is no interference, the maximum achievable SIR is unbounded. If there is only one slot ($S = 1$), the maximum achievable SIR for these B users is equal to the reciprocal of the largest eigenvalue of a $B \times B$ matrix F. F is closely related to the matrix $H(\vec{\gamma}_1)$ and has diagonal elements equal to zero and the off diagonal element (i, j) equal to G_{ij} / G_{ii} [7].

(6) is a set of non-linear equations with $2 \times B$ unknowns: γ_{ik} and p_{ik} for each slot k. We can cut the number of variables in half by applying the Perron-Frobenius theorem [11][12]. The theorem states that in order for (6) to have non-negative feasible power solution, the necessary and sufficient condition is that

$$\Delta_{i,k} \ge 0 \tag{7}$$

where $\Delta_{i,k}$ is the i-th principle minors of $H(\vec{\gamma}_k)$ and $1 \le i \le B$. In other words, the determinants of all of the principle submatrices of $H(\vec{\gamma}_k)$ along the diagonal have to be greater than or equal to zero. Since $\Delta_{i,k}$ is only a function of the desired SIR, γ_{ik}, replacing (6) by (7) removes all of the variables representing transmitted power, p_{ik}, from the problem.

Even though the cost function (4) is a linear function and the feasible region for (5) is convex, the optimization problem is unfortunately NP-hard. There are no algorithms that can solve this problem in polynomial time. To prove the optimization problem is NP-hard, we only need to show that the feasible region for one of the constraints is non-convex. The proof is straight-forward. We will show that the feasible region for the 2×2 principle minor, Δ_{2k}, constraint is non-convex.

We expand the 2×2 principle matrix from (7) and drop the slot index k for simplicity in notation. We assume that there are two sets of SIR (γ_1^1, γ_2^1) and (γ_1^2, γ_2^2) that lie on the boundaries of the feasible region:

$$\begin{aligned} 1 - \gamma_1^1 \gamma_2^1 G_{12} G_{21} / G_{11} G_{22} &= 0 \\ 1 - \gamma_1^2 \gamma_2^2 G_{12} G_{21} / G_{11} G_{22} &= 0 \end{aligned} \tag{8}$$

Let (γ_1^1, γ_2^1) and (γ_1^2, γ_2^2) be distinct and nonzero. We would like to show that for some positive number a, $0 < a < 1$, $(a\gamma_1^1 - (1-a)\gamma_1^2, a\gamma_2^1 - (1-a)\gamma_2^2)$ is not in the feasible region. That is, we need to show

$$1 - [a\gamma_1^1 - (1-a)\gamma_1^2]\,[a\gamma_2^1 - (1-a)\gamma_2^2]\, G_{12} G_{21} / G_{11} G_{22} \tag{9}$$

is less than zero. After some algebraic manipulation and substitution using (8), (9) is equal to

$$a(1-a)\left[2 - \left(\frac{\gamma_2^1}{\gamma_2^2} - \frac{\gamma_2^2}{\gamma_2^1}\right)\right]$$

Since $\gamma_2^1 > 0$ and $\gamma_2^2 > 0$,

$$\left(\frac{\gamma_2^1}{\gamma_2^2} - \frac{\gamma_2^2}{\gamma_2^1}\right) \geq 2$$

and is equal two only if $\gamma_2^1 = \gamma_2^2$ which violates our initial assumption of $(\gamma_1^1, \gamma_2^1) \neq (\gamma_1^2, \gamma_2^2)$. Since a and (1 - a) are positive, (9) is less than zero. The feasible region for the 2×2 principle minor is non-convex and the optimum bit allocation problem (4) is NP-hard. It is also possible to prove that the optimization problem is NP-hard without using the Perron-Frobenius theorem [6].

4 Bit Allocation Algorithms

In this section, we propose several *ad hoc* bit allocation algorithms derived from the insights obtained from exhaustive searches [6]. Dynamic simulations for each algorithm are performed for the reverse link of a nineteen-cell indoor wireless system. QAM is used. The minimum constellation is 4-QAM with desired SIR of 10 dB. An additional 3 dB of SIR is needed for every additional bit. The signals are corrupted by additive white Gaussian noise. We impose an additional constraint for the maximum constellation due to hardware linearity constraints as set 64-QAM as the largest constellation possible. We assume the users arrive with a Poisson distribution and have i.i.d. exponential service time. The FH-CDMA systems have 23 frequency bins. The period of all hopping patterns is equal to N=23 time slots and is

conveniently defined as a frame. There are twenty three hopping patterns available in every cell. Each slot has C symbols. We assume every user desires $50 \times C$ bits/frame. We further assume that the users and surrounding objects are stationary so that the channel conditions do not change for the duration of the calls.

We compare six different bit allocation algorithms. They are:

1. minimizing the total transmitted power in the hopping pattern
2. minimizing the maximum transmitted power in the hopping pattern
3. minimizing the total interference induced in the hopping pattern
4. minimizing the maximum of the interference induced in the hopping pattern
5. minimizing the total number of slots used in the hopping pattern
6. reverse water-filling of SIR

Algorithms 1 and 2 are both transmit power based. Algorithms 3 and 4 are interference based. Algorithm 5 performs interference avoidance by limiting the number of slots used. It is designed to give the users arrive later more freedom to perform interference avoidance. Algorithm 6 performs reverse water-filling of SIR. Reverse water-filling of SIR can be best described by turning Figure 1, which shows the maximum achievable SIR for all available slots, upside-down. A new user first determines the maximum achievable SIR, γ_{max}, in each slot via channel probing. SIR is then "poured" into these slots with uneven "depth" until the desired throughput is achieved with SIR of γ'. Reverse water-filling of SIR is the optimum bit allocation for the single-user multitone system at high SIR [2].

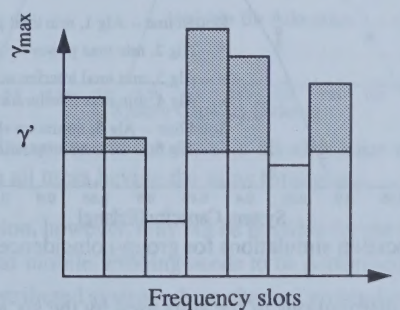


Figure 1 Reverse water-filling of SIR for a multislot system.

We present the simulation results for indoor FH-CDMA wireless systems group-coincidence hopping (GC) patterns. The group-coincidence FH-CDMA systems assign a new user to share every slot in its hopping pattern with one group of users. The group of potential interferers is the same is every hop. Depending on how the users in the system allocate their bit rates, a new user may not share every slot with all of the users of the same hopping pattern. There is no *intra-cell* interference as the users do not share any slot in their hopping patterns with the other users in the same cells.

Before presenting the simulation results, we define the system capacity in units of Erlangs as the average arrival rate divided by the total number of hopping patterns available in

the system which is equal to the number of base stations multiplied by the number of hopping patterns available in each cell. We assume that once a user is admitted into the system, it keeps the bit assignments until it terminates the call. No users are dropped due to the arrivals of new users. It is up to the new users to adjust their channel and bit assignments to the different interference levels. Centralized power control is used for our simulations to reduce simulation time.

Figure 2 shows the probability that a new arrival is blocked, P_{block}, as a function of system capacity. It shows that the transmitted power-based Algorithms 1 and 2 have the lowest system capacity. Algorithm 5, which minimizes the number of slots a new user occupies, achieves the highest capacity among the six algorithms. The interference-based Algorithms 3 and 4 and the reverse water-filling of SIR, Algorithm 6, all have similar system capacities and achieve capacity close to Algorithm 5. The difference in system capacities for Algorithm 5 and Algorithm 1 for P_{block} of 0.1% to 1% is approximately 30% to 45%.

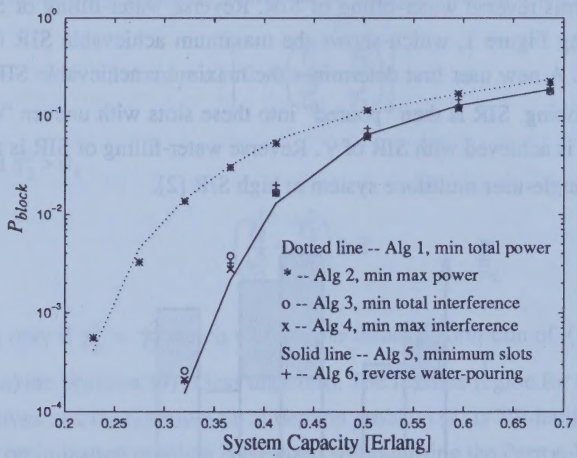


Figure 2 Bit allocation simulations for group-coincidence systems.

Figure 3 shows the distributions of the slots used by the six algorithms as a function of the bits carried per symbol. No slots carry just one bit of information as BPSK is not allowed. Algorithm 5, which achieves the highest system capacity, is able to assign the maximum constellation of 64-QAM to the most slots among the six algorithms and at the same time leave the most slots unused. It minimizes the number of slots a new user transmits and concentrates its throughput and power in these slots. The new user avoids the slots with large interference when possible. These are the characteristics associated with the maximum throughput bit assignments found by the exhaustive searches as described in [6]. Minimizing the transmitted slots to which a user allocates power also gives future arrivals more flexibility to perform interference avoidance.

4. The system with M/N = 8/23 can obtain higher system capacity than the M/N = 12/23. The users in the M/N = 8/23 system concentrate their throughput in fewer slots and is more effective in avoiding large interferers. Figure 4 also shows that the system capacity of the variable bit allocation system is 50%-64% better than that of the M/N= 8/23 system for P_{block} of 0.1% to 1% and 75% to 100% better than that of the M/N = 12/23 system. The bit distribution among the slots of the variable bit allocation system is similar to Figure 3 for Algorithm 5. The largest constellations, which are the same as the M/N=8/23 system, are used in most of the slots that carry information. The bit allocation system, however, has the additional flexibility to allocate bits according to the congestion condition in the hopping patterns. It can admit users that are unable to achieve the fixed constellation in every slot used as in the M/N = 8/23. The system capacity improves significantly.

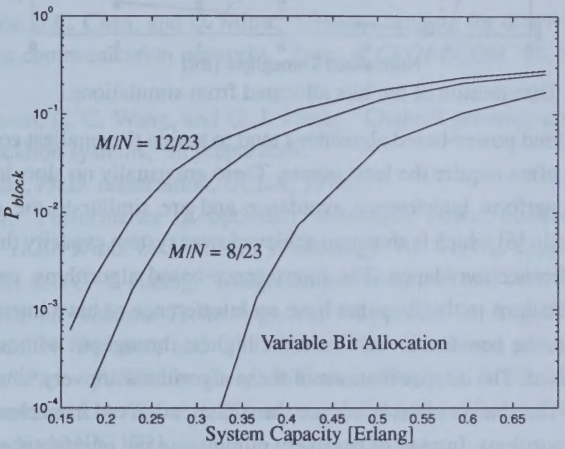


Figure 4 Simulation results for variable bit allocation and M/N systems with all users having the same throughput.

Variable bit allocation, however, may not be possible for the wireless environments that change quickly, e.g. cellular mobile. Probing needs to be performed constantly under the fast-varying conditions. In a distributed system where channel probing and power control take time to adapt, the delay, along with the time it takes to calculate the new bit allocations, may render the new bit allocations difficult or even impossible to obtain as the channels vary.

5 Conclusion

Finding the bit allocation that maximizes the system throughput is difficult. In an FH-CDMA system where every user requires the same bit rate, we propose several *ad hoc* algorithms that utilize the insights we have obtained from exhaustive searches. Interference avoidance which asks the users to avoid transmitting in the slots with large interferers nearby seems to be the best strategy. These bit allocation algorithms gives the system more flexibility to adapt to the interference statistics and result in the highest system capacity.

In our simulations, we assume that power control and bit allocation can be performed

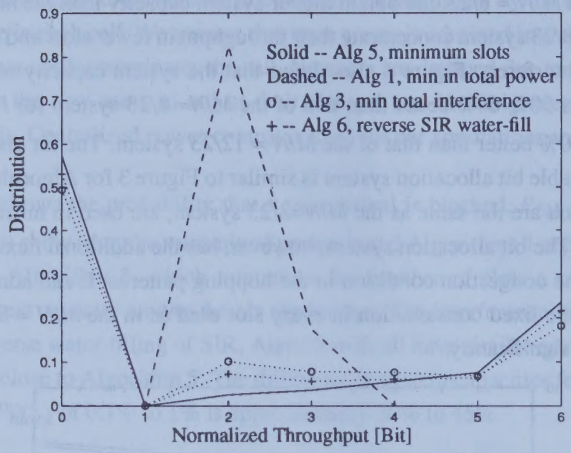


Figure 3 Distribution of the bits allocated from simulations.

The transmitted power-based algorithms tend to assign the smallest constellations to all of the slots as they often require the least power. There are usually no slots left unused. These algorithms do not perform interference avoidance and are similar to the equal throughput algorithm described in [6] which is shown to achieve lower system capacity than the algorithms that perform interference avoidance. The interference-based algorithms, on the other hand, assign large constellations to the slots that have no interference or have interferers that are far away. In these slots, the new users can obtain the highest throughput without inducing much additional interference. The bit distributions of these algorithms are very similar to Algorithm 5. The interference-based algorithms better use the statistics derived from channel probing than the power-based algorithms. Instead of passively minimizing the interference generated by the new user as in Algorithms 1 and 2, Algorithms 3 and 4 estimate how the users already in the system would react to the interference generated by the new user. Algorithm 6, which performs reverse water-filling of SIR, also obtains the most throughput desired in very few slots. These slots usually have very high achievable SIR and on the average, have no interferers or have interferers that have very low mutual link gain with the new user. The bit distribution of Algorithm 6 also takes on the general shape of Algorithm 5.

We now compare the system capacity of the variable bit FH-CDMA systems with the FH-CDMA systems deploying interference avoidance in the form of transmitting in M out of every N slots as in [13]. N is set to twenty-three. The M/N systems assign a fixed constellation in the M slots they view as favorable. We assume that these systems are operating in the same environment and have the same bit rate of $48 \times C$ bits/frame where C is the number of symbols per slot. The variable bit allocation system utilizes the minimum-slot Algorithm 5. The variable bit allocation can carry as much as six bits of information in one symbol. Two M/N systems are simulated. In one of the systems, each symbol transmitted carries four bits of information. Twelve out of every twenty-three slots are used (M/N =12/23). In the other M/N system, every symbol transmitted carries six bits of information (M/N = 8/23). The results are shown in Figure

instantly. We have not taken into the account of the speed of convergence of various adaptive algorithms. The results presented here are meant to serve as an upperbound on how the various proposed bit allocation algorithms can improve the system's capacity.

Reference

[1] J. A. C. Bingham, "Multicarrier modulation for data transmission: an idea whose time has come," *IEEE Comm.*, pp. 5-14, May, 1990.

[2] D. Hughes-Hartogs, United Stated Patent 4,679,227. Jul. 1987.

[3] C. J. Hansen, C. C. Wang, and G. J. Pottie, "Distributed dynamic channel resource allocation in wireless systems," *Proc. of Asilomar Conference on Signals, Systems, and Computers*, Oct. 1994.

[4] N. Bambos, S.C. Chen, and D. Mitra, "Channel probing for distributed access control in wireless communication networks," *Proc. of GLOBECOM '95*, Singapore, pp. 322-6.

[5] C. J. Hansen, C. C. Wang, and G. J. Pottie, "Channel probling algorithms in wirelss communication systems," in preparation.

[6] C. C. Wang, Ph.D. dissertation, UCLA, 1996.

[7] J. Zander, "Performance of optimum transmitter power control in cellular radio systems," *IEEE Trans. on Vehicular Technology*, Vol. 41, No. 1, pp. 57-63, Feb. 1992.

[8] R. D. Yates and C.-Y. Huang, "Integrated power control and base station assignment," *IEEE Trans. on Vehicular Technology*, Vol. 44, pp. 538-44, Aug. 1995.

[9] S. C. Chen, N. Bambos, and G. J. Pottie, "On distributed power control for radio networks," *Proc. of IEEE Int. Conf. on Comm* 1994, pp. 1281-5, May 1994.

[10] T. M. Cover and J. A. Thomas, *Elements of Information Theory*, John Wiley and Sons, Inc., New York, NY, 1991.

[11] F. R. Gantmakher, *The Theory of Matrices*, Chelsea Pub. Co., New York, 1959.

[12] E. Senata, *Non-Negative Matrices*. Wiley and Sons: New York, 1973.

[13] C. C. Wang and G. J. Pottie, "Interference avoidance and power control strategies for coded frequency hopped cellular systems," *Proc. of IEEE Int. Conf. on Comm.*, June 1995, pp. 1737-41. Seattle, WA.

Quantum Error Correction Via Codes Over $GF(4)$

A. R. Calderbank,[1] E. M. Rains,[2] P. W. Shor,[1] and N. J. A. Sloane[1]

[1]AT&T Research, Murray Hill, New Jersey 07974
[2]Institute for Defense Analyses, Princeton, New Jersey 08540

Abstract. The problem of finding quantum-error-correcting codes is transformed into the problem of finding additive codes over the field $GF(4)$ which are self-orthogonal with respect to a certain trace inner product.

1. Introduction

The relationship between quantum information and classical information is a subject currently receiving much study. While there are many similarities, there are also substantial differences between the two. Classical information cannot travel faster than light, while quantum information appears to in some circumstances. Classical information can be duplicated, while quantum information cannot [8], [16].

It is well known that classical information can be protected from degradation by the use of classical error-correcting codes [11]. Classical error-correcting codes appear to protect classical information by duplicating it, so because of the theorem that a quantum bit cannot be cloned, it was widely believed that these techniques could not be applied to quantum information. That quantum-error-correcting codes could indeed exist was recently shown by one of us [12]. Two of us [7] then showed that a class of good quantum codes could be obtained by using a construction that starts with a binary linear code C containing its dual $C^{\perp}$. Independently, Steane also discovered the existence of quantum codes [15] and the same construction [14]. At around the same time, Bennett et al. [2] discovered that two experimenters each holding one component of many noisy Einstein-Podolsky-Rosen (EPR) pairs could *purify* them, yielding fewer nearly perfect EPR pairs. The resulting pairs can then be used to teleport quantum information from one experimenter to the other [1]. Although it was not immediately apparent, these two discoveries turned out to be different ways of looking at the same phenomenon. A purification protocol that uses only one-way classical communication between the experimenters can be converted into a quantum-error-correcting code, and vice versa [3]. After these discoveries, a number of improved quantum codes were soon found by various

researchers.

The setting in which quantum-error-correcting codes exist is the quantum state space of n qubits (quantum bits, or two-state quantum systems). This space is $\mathbb{C}^{2^n}$, and it has a natural decomposition as the tensor product of n copies of $\mathbb{C}^2$, where each copy corresponds to one qubit. The known quantum codes seemed to have close connections to a finite group of unitary transformations of $\mathbb{C}^{2^n}$, known as a Clifford group, and denoted here by L. This group contains all the transformations necessary for encoding and decoding quantum codes. It is also the group generated by fault-tolerant bitwise operations performed on qubits that are encoded by certain quantum codes [7], [13], [7], [14]. Investigation of the connection between this group and existing quantum codes led to a general construction for such codes which allowed us to generate many new examples. The initial results of this study were reported in [5], and the work is continued in the present paper.

2. From quantum codes to binary spaces

A quantum error-correcting code is a way of encoding quantum states into qubits so that errors or decoherence in a small number of individual qubits have little or no effect on the encoded data. A bit error in an individual qubit corresponds to applying the Pauli matrix $\sigma_x = \left(\begin{smallmatrix} 0 & 1 \\ 1 & 0 \end{smallmatrix}\right)$ to that qubit, and a phase error to the Pauli matrix $\sigma_z = \left(\begin{smallmatrix} 1 & 0 \\ 0 & -1 \end{smallmatrix}\right)$. The group $E \subset O(2^n)$ of tensor products $\pm w_1 \otimes \cdots \otimes w_n$, where each w_i is one of I, σ_x, σ_z, $\sigma_x\sigma_z$, describes the possible errors in n qubits. For the purposes of quantum error correction, we need consider only the three types of errors σ_x, σ_z and $\sigma_x\sigma_z$, since any error-correcting code which corrects t of these errors will be able to correct arbitrary errors in t qubits [3], [9]. Our codes will thus be tailored for the error model in which each qubit undergoes independent errors, and the three errors σ_x, σ_z and $\sigma_x\sigma_z$ are all equally likely. The results of [3], [9] show that any code which corrects these types of quantum errors will be able to correct errors in arbitrary error models, assuming the errors are not correlated among large numbers of qubits and that the error rate is small. It may be possible for other error models to find codes which correct errors more efficiently than our codes do; this is not discussed in this paper.

The group E is an extraspecial 2-group with center $\Xi(E) = \{\pm I\}$. The extraspecial condition is that $\overline{E} = E/\Xi(E)$ is elementary abelian, and hence a binary vector space.

Let V denote the vector space $\mathbb{Z}_2^n$ (where $\mathbb{Z}_2 = \{0,1\}$), and label the standard basis of $\mathbb{C}^{2^n}$ by $|v\rangle$, $v \in V$. Every element $e \in E$ can be written uniquely in the form

$$e = X(a)Z(b)(-I)^\lambda \tag{1}$$

where $\lambda \in \mathbb{Z}_2$, $X(a) : |v\rangle \to |v+a\rangle$, $Z(b) : |v\rangle \to (-1)^{b\cdot v}|v\rangle$, for $a, b \in V$. The element $X(a)Z(b)$ indicates that there are bit errors in the qubits for which $a_j = 1$ and phase errors in the qubits for which $b_j = 1$.

If $e \in E$ is given by (1), the corresponding element $\overline{e} \in \overline{E}$ will be written $(a|b)$. There is a symplectic inner product $\overline{E} \times \overline{E} \to \mathbb{Z}_2$ given by

$$((a|b),(a'|b')) = a \cdot b' + a' \cdot b \; . \tag{2}$$

If $e, e' \in E$ then $ee' = \pm e'e$, where the sign is given by

$$\{X(a)Z(b)\}\{X(a')Z(b')\} = (-1)^{a\cdot b' + a'\cdot b}\{X(a')Z(b')\}\{X(a)Z(b)\} \; . \tag{3}$$

A subspace $\overline{S}$ of $\overline{E}$ is said to be totally isotropic if for all $\overline{s}_1$, $\overline{s}_2 \in \overline{S}$ the symplectic inner product $(\overline{s}_1, \overline{s}_2) = 0$. A subgroup S of E is commutative if and only if its image $\overline{S}$ in $\overline{E}$ is totally isotropic. The dimension of a totally isotropic subspace is at most n. The groups $X = \{X(a) : a \in V\}$ and $Z = \{Z(b) : b \in V\}$ are examples of subgroups of E whose images $\overline{X}$, $\overline{Z}$ have dimension n.

Since Z is a group of $2^n \times 2^n$ commuting symmetric matrices, these matrices can be simultaneously diagonalized. The one-dimensional eigenspaces are spanned by eigenvectors $|v\rangle$, $v \in V$ and each such space affords a different linear character of Z. Similarly, the eigenvectors for the commutative subgroup X have the form

$$f_w = 2^{-n/2} \sum_{v \in V} (-1)^{w\cdot v}|v\rangle, \quad w \in V \; , \tag{4}$$

and afford the 2^n different linear characters of X.

We consider the group $E' \subset U(2^n)$ generated by E and iI. The center $\Xi(E')$ is $\langle iI \rangle$, so that $E'/\Xi(E') \cong E/\Xi(E) = \overline{E}$. The normalizer L of E' in $U(2^n)$ acts on E' by conjugation, fixing the center $\Xi(E')$, so that there is a well-defined action of L on $\overline{E}$ that preserves the symplectic inner product (see [4] for details).

The group-theoretic construction of quantum-error-correcting codes given in [5] starts by choosing a commutative subgroup S of E' that intersects the center $\Xi(E')$ in $\langle I \rangle$. The corresponding subspace $\overline{S} \subset \overline{E}$ is totally isotropic, and if $\dim \overline{S} = k$, then $\overline{S}$ has 2^k distinct

linear characters, each afforded by a 2^{n-k}-dimensional eigenspace of $\mathbb{C}^{2^n}$. We choose one of these eigenspaces, which without loss of generality we can take to be the eigenspace R corresponding to the trivial character. Then R is a quantum-error-correcting code that encodes $n-k$ qubits into n qubits. The quantum-error-correcting properties of R are determined by combinatorial properties of S.

For simplicity we assume for the moment that the commutative subgroup S is disjoint from Z, so that it is contained in an n-dimensional commutative subgroup W disjoint from Z. The group W can be obtained by conjugating X by a suitable diagonal matrix $d_P = \mathrm{diag}\{i^{T_P(v)}\}$, where $T_P : V \to \mathbb{Z}_4 = \mathbb{Z}/4\mathbb{Z}$ is a $\mathbb{Z}_4$-valued quadratic form associated with the binary symmetric matrix P. Thus

$$T_P(u+v) = T_P(u) + T_P(v) + 2uPv^{tr} \ , \tag{5}$$

where the entries 0, 1 of P are regarded as elements of $\mathbb{Z}_4$. The effect on $\overline{E}$ of conjugation by d_P is given by $\left(\begin{smallmatrix} I & P \\ 0 & I \end{smallmatrix}\right)$; this means that $\overline{X}$, the subspace of $\overline{E}$ of elements of the form $(a|0)$, $a \in V$, is mapped to

$$\overline{W} = d_P^{-1}\overline{X}d_P = \{(a|aP) : a \in V\} \ .$$

The eigenvector f_w for X is mapped to the eigenvector $f_w d_P$ for W. For more details see Section 4 of [4]. Continuing to assume that S is disjoint from Z, we may suppose that $\overline{S}$ is generated by the first k rows of the matrix $[I|P]$, so that

$$\overline{S} = \{(a|aP) : a = (a_1, \ldots, a_n) \in V \ \text{ with } \ a_{k+1} = \cdots = a_n = 0\} \ .$$

We now determine the eigenspace R corresponding to the trivial character. We calculate

$$\begin{aligned}
2^{n/2}(f_w d_P)X(a)Z(aP) &= \sum_v i^{2w\cdot v + T_P(v)} \ |v+a\rangle \ Z(aP) \\
&= \sum_v i^{2w\cdot v + T_P(v) + 2(v+a)Pa^{tr}} |v+a\rangle \\
&= \sum_v i^{2w\cdot(v+a) + T_P(v+a) + 2vPa^{tr}} |v\rangle \\
&= (-1)^{w\cdot a} \sum_v i^{T_P(v)+T_P(a)+2w\cdot v} |v\rangle \\
&= i^{T_P(a)}(-1)^{w\cdot a} \left(\sum_v (-1)^{w\cdot v} |v\rangle \right) d_P \\
&= 2^{n/2} i^{T_P(a)} (-1)^{w\cdot a} f_w d_P \ ,
\end{aligned} \tag{6}$$

where we have made use of (4) and (5). In other words

$$(f_w d_P)\overline{X(a)\,Z(aP)} = (-1)^{w\cdot a}(f_w d_P)\ . \tag{7}$$

It follows from (7) that the linear character of $\overline{S}$ afforded by $f_w d_P$ is $w \to (-1)^{w\cdot a}$. The sign $(-1)^{w\cdot a}$ may be obtained directly by applying $i^{-T_P(a)}X(a)Z(aP)$ to the vector $f_w d_P$. Clearly any $w = (w_1, \ldots, w_n)$ with $w_1 = \cdots = w_k = 0$ determines the trivial character. Thus the quantum-error-correcting code R is the subspace of $\mathbb{C}^{2^n}$ spanned by the vectors $f_w d_P$ with $w_1 = \cdots = w_k = 0$.

Now suppose that $r \in R$ is subjected to an error process $X(a)Z(b)$. The vector $r' = rX(a)Z(b)$ is in $R' = RX(a)Z(b)$, which is an eigenspace of S corresponding to some linear character λ, possibly not the trivial character. We need to identify the eigenspace indirectly (not by measuring v') so that we can invert the error process. That is, we want to identify the eigenspace without measuring which $r \in R$ was subjected to the error process. For any $a' \in V$ we have

$$\begin{aligned}\{rX(a)Z(b)\}X(a')Z(a'P) &= (-1)^{a'Pa^{tr}+a'\cdot b}\{rX(a')Z(a'P)\}\{X(a)Z(b)\} \\ &= (-1)^{a'[I|P](b|a)^{tr}} i^{T_P(a')} rX(a)Z(b)\ . \end{aligned} \tag{8}$$

It is easy to show that $i^{T_P(a')}X(a')Z(a'P)$ is hermitian, so the sign in (8) is observable. Further, for all $a' \in V$, the matrices $X(a')Z(a'P)$ commute, so for different $a' \in V$, these signs are simultaneously observable. By taking a' to be each of the first k unit vectors in turn we obtain the syndrome of the vector $(b|a)$ with respect to a parity check matrix for $S^{\perp}$, the dual space to S with respect to the symplectic inner product (2).

Finally, we describe more explicitly how the error process $X(a)Z(b)$ is identified. The problem is that of identifying the character afforded by the eigenspace $R' = RX(a)Z(b)$. For simplicity we begin by taking an element $Z(b') \in S$ and looking at its effect on R'. We take $r \in R$ and consider

$$r' = rX(a)Z(b) = \sum_{v\in V} \alpha_v |v\rangle\ ,$$

say, when $\alpha_v \in \mathbb{C}$. Since r is an eigenvector for $Z(b')$ with eigenvalue $\epsilon = \pm 1$,

$$r'Z(b') = \{rX(a)Z(b)\}Z(b') = \epsilon(-1)^{a\cdot b'}\{rX(a)Z(b)\}\ .$$

This means that $(-1)^{b'\cdot v}$ is constant over all vectors v for which $\alpha_v \neq 0$. We can calculate this sign as follows. Suppose for concreteness that $b'\cdot v = \sum_{i=1}^{n} b'_i v_i = v_1 + v_2 + v_4$. We

may compute this by taking an ancillary qubit initialized to $|0\rangle$ and applying a succession of XOR's:

$$\begin{aligned}\left(\sum_v \alpha_v |v_1 \dots v_n\rangle\right)|0\rangle &\rightarrow \sum_v \alpha_v |v_1 \dots v_n (v_1 + 0)\rangle \\ &\rightarrow \sum_v \alpha_v |v_1 \dots v_n (v_1 + v_2)\rangle \\ &\rightarrow \sum_v \alpha_v |v_1 \dots v_n (v_1 + v_2 + v_4)\rangle\end{aligned}$$

To identify the effect of a general element $X(a')Z(b')$ of S, we use the fact that the group L acts transitively on nonzero vectors: there is an element $g \in L$ such that

$$g^{-1}X(a')Z(b')g = Z(b') \ .$$

This g is a unitary transformation (which preserves quantum information) so the effect of applying $X(a')Z(b')$ to R' is the same as that of applying $Z(b')$ to $R'g$.

The above discussion can be summarized as follows. Let $\overline{E}$ denote a $2n$-dimensional binary vector space, whose elements are written $(a|b)$ and which is equipped with the inner product (2). Define the weight of $(a|b) = (a_1 \cdots a_n | b_1 \cdots b_n)$ to be the number of coordinates i such that at least one of a_i and b_i is 1. The distance between two elements $(a|b)$, $(a'|b') \in \overline{E}$ is defined to be the weight of their difference.

Then we have the following theorem, which is an immediate consequence of Theorem 1 of [5].

Theorem 1. *Suppose $\overline{S}$ is a k-dimensional linear subspace of $\overline{E}$ which is contained in its dual $\overline{S}^{\perp}$ (with respect to the inner product (2)), and is such that there are no vectors of weight $\leq d-1$ in $\overline{S}^{\perp} \setminus \overline{S}$. Then by taking as codewords all the 2^{n-k} vectors in some eigenspace (for any chosen linear character) of $\overline{S}$, we obtain a quantum-error-correcting code mapping $n-k$ qubits to n qubits which can correct $[(d-1)/2]$ errors (and if d is even can also detect $d/2$ errors).*

We will describe such a quantum-error-correcting code by saying it has parameters $[[n, n-k, d]]$. All quantum-error-correcting codes known at the present time can be constructed in this way.

Example. We describe the error-correcting code mapping 1 qubit into 5 qubits presented in [3]. This code contains two codewords.

$$\begin{aligned}|c_0\rangle = &|00000\rangle \\ &+ |11000\rangle + |01100\rangle + |00110\rangle + |00011\rangle + |10001\rangle \\ &- |10100\rangle - |01010\rangle - |00101\rangle - |10010\rangle - |01001\rangle \\ &- |11110\rangle - |01111\rangle - |10111\rangle - |11011\rangle - |11101\rangle \ ,\end{aligned}$$

and

$$\begin{aligned}|c_1\rangle = &|11111\rangle \\ &+ |00111\rangle + |10011\rangle + |11001\rangle + |11100\rangle + |01110\rangle \\ &- |01011\rangle - |10101\rangle - |11010\rangle - |01101\rangle - |10110\rangle \\ &- |00001\rangle - |10000\rangle - |01000\rangle - |00100\rangle - |00010\rangle \ .\end{aligned}$$

Essentially the same code is given in [10], but we use the presentation above as it is symmetric under cyclic permutations. If $X(11000)Z(00101)$ is applied to $|c_0\rangle$ and $|c_1\rangle$, it is easily verified that these codewords are left invariant. Thus, the vector $(11000|00101) \in \overline{E}$ is one of the vectors in the subspace $\overline{S}$. Since the code is symmetric under cyclic permutations, four other vectors of $\overline{S}$ is found, and it can easily be shown that the four vectors

11000	00101
01100	10010
00110	01001
00011	10100

generate a 4-dimensional totally singular subspace $\overline{S}$. [The fifth cyclic shift, (10001|01010), is in the subspace generated by these four.] The dual $\overline{S}^{\perp}$ of $\overline{S}$ is generated by $\overline{S}$ and the additional two vectors (11111|00000) and (00000|11111). It is straightforward to verify that the minimum weight vector in $\overline{S}^{\perp}$ has weight three [one of these vectors is (00111|00101)] and thus this code can correct one error.

3. From binary spaces to codes over $GF(4)$

As is customary (cf. [11]) we take the Galois field $GF(4)$ to consist of the elements $\{0, 1, \omega, \overline{\omega}\}$, with $\overline{\omega} = \omega^2 = \omega + 1$, $\omega^3 = 1$; the trace map $\mathrm{Tr} : GF(4) \to \mathbb{Z}_2$ takes $x \in GF(4)$ to $x + x^2$. The *Hamming weight* of a vector $u \in GF(4)^n$, written $\mathrm{wt}(u)$, is the number of nonzero components, and the *Hamming distance* between $u, u' \in GF(4)^n$ is $\mathrm{dist}(u, u') = \mathrm{wt}(u - u')$. The minimal Hamming distance between the members of a subset C of $GF(4)^n$ will be denoted by $\mathrm{dist}(C)$.

To each vector $v = (a|b) \in \overline{E}$ we associate the vector $\phi(v) = \omega a + b \in GF(4)^n$. It is immediate that the weight of v is equal to the Hamming weight of $\phi(v)$, and the distance

between vectors $v = (a|b)$, $v' = (a'|b') \in \overline{E}$ is equal to $\mathrm{dist}(\phi(v), \phi(v'))$. The symplectic inner product of v and v' (see (2)) is equal to $\mathrm{Tr}(\phi(v) \cdot \overline{\phi(v')})$, where the bar denotes conjugation in $GF(4)$, since

$$\begin{aligned} \mathrm{Tr}(\phi(v) \cdot \overline{\phi(v')}) &= \mathrm{Tr}((\omega a + b) \cdot (\overline{\omega} a' + b')) \\ &= (a \cdot a')\mathrm{Tr}(1) + (a \cdot b')\mathrm{Tr}(\omega) + (a' \cdot b)\mathrm{Tr}(\overline{\omega}) + (b \cdot b')\mathrm{Tr}(1) \\ &= a \cdot b' + a' \cdot b \, . \end{aligned}$$

If $\overline{S}$ is a linear subspace of $\overline{E}$ then $C = \phi(\overline{S})$ is a subset of $GF(4)^n$ which is closed under addition. We shall refer to C as an *additive* code over $GF(4)$, and refer to it as an $(n, 2^k)$ code if it contains 2^k vectors. If C is also closed under multiplication by ω, we say it is *linear*.

The *trace inner product* of vectors $u, v \in GF(4)^n$ will be denoted by

$$u * v = \mathrm{Tr}\, u \cdot \overline{v} = \sum_{j=1}^{n} u_j \overline{v}_j + \left(\sum_{j=1}^{n} u_j \overline{v}_j \right)^2 . \tag{9}$$

If C is an $(n, 2^k)$ additive code, its dual is defined to be

$$C^{\perp} = \{ u \in GF(4)^n : u * v = 0 \ \text{ for all } \ v \in C \} \, . \tag{10}$$

Then $C^{\perp}$ is an $(n, 2^{2n-k})$ code. If $C \subseteq C^{\perp}$ we say C is *self-orthogonal*, and if $C = C^{\perp}$ then C is *self-dual*.

Theorem 1 can now be reformulated.

Theorem 2. *Suppose C is an additive self-orthogonal subcode of $GF(4)^n$, containing 2^k vectors, such that there are no vectors of weight $\leq d-1$ in $C^{\perp} \setminus C$. Then any eigenspace for $\phi^{-1}(C)$ is a quantum-error-correcting code with parameters $[[n, n-k, d]]$.*

We say that C (or the associated quantum-error-correcting code) is *nondegenerate* if there are no nonzero vectors of weight $\leq d-1$ in $C^{\perp}$; otherwise we call C *degenerate*. We say that a quantum-error-correcting code is *linear* if the associated additive code C is linear.

4. A table of quantum-error-correcting codes

The full journal paper [6] contains a great many explicit constructions for quantum error correcting codes and a linear programming upper bound. The table below shows

our present state of knowledge about the highest minimum distance d in any $[[n, k, d]]$ code where $n \leq 30$ and $k \leq 7$.

Table I: Highest achievable minimal distance d in any $[[n, k, d]]$ quantum-error-correcting code.

$n \setminus k$	0	1	2	3	4	5	6	7
3	2	1	1	1				
4	2	2	2	1	1			
5	3	3	2	1	1	1		
6	4	3	2	2	2	1	1	
7	3	3	2	2	2	1	1	1
8	4	3	3	3	2	2	2	1
9	4	3	3	3	2	2	2	1
10	4	4	4	3	3	2	2	2
11	5	5	4	3	3	3	2	2
12	6	5	4	4	4	3	3	2
13	5	5	4	4	4	3 – 4	3	3
14	6	5	4 – 5	4 – 5	4	4	4	3
15	6	5	5	5	4	4	4	3
16	6	6	6	5	4 – 5	4 – 5	4	3 – 4
17	7	7	6	5 – 6	4 – 5	4 – 5	4 – 5	4
18	8	7	6	5 – 6	5 – 6	5	5	4
19	7 – 8	7	6	5 – 6	5 – 6	5 – 6	5	4 – 5
20	8	7	6 – 7	5 – 7	5 – 6	5 – 6	5 – 6	4 – 5
21	8	7	6 – 7	5 – 7	5 – 7	5 – 6	5 – 6	4 – 6
22	8	7 – 8	6 – 8	5 – 7	5 – 7	5 – 7	5 – 6	4 – 6
23	8 – 9	7 – 9	6 – 8	5 – 8	5 – 7	5 – 7	5 – 7	5 – 6
24	8 – 10	8 – 9	6 – 8	6 – 8	6 – 8	6 – 7	6 – 7	5 – 7
25	8 – 9	9	7 – 8	7 – 8	7 – 8	7 – 8	6 – 7	5 – 7
26	8 – 10	9	8 – 9	8 – 9	8	7 – 8	6 – 8	5 – 8
27	9 – 10	9	9	9	8 – 9	7 – 8	6 – 8	5 – 8
28	10	10	10	9	8 – 9	7 – 9	6 – 8	6 – 8
29	11	11	10	9 – 10	8 – 9	7 – 9	6 – 9	6 – 8
30	12	11	10	9 – 10	8 – 10	7 – 9	6 – 9	6 – 9

References

[1] C. H. Bennett, G. Brassard, C. Crépeau, R. Jozsa, A. Peres and W. K. Wooters, "Teleporting an unknown quantum state via dual classical and EPR channels," *Phys. Rev. Lett.*, **70**, pp. 1895–1898 (1993).

[2] C. H. Bennett, G. Brassard, S. Popescu, B. Schumacher, J. A. Smolin and W. K. Wooters, "Purification of noisy entanglement and faithful teleportation via noisy

channels," *Phys. Rev. Lett.*, **76**, pp. 722–725 (1996).

[3] C. H. Bennett, D. DiVincenzo, J. A. Smolin and W. K. Wooters, "Mixed state entanglement and quantum error correction," *Phys. Rev. A*, submitted; also LANL e-print quant-ph/9604024.

[4] A. R. Calderbank, P. J. Cameron, W. M. Kantor and J. J. Seidel, "$\mathbb{Z}_4$ Kerdock codes, orthogonal spreads, and extremal Euclidean line-sets," *Proc. London Math. Soc.* (to appear).

[5] A. R. Calderbank, E. M. Rains, P. W. Shor and N. J. A. Sloane, "Quantum error correction and orthogonal geometry," *Phys. Rev. Lett.*, submitted.

[6] A. R. Calderbank, E. M. Rains, P. W. Shor and N. J. A. Sloane, "Quantum error correction via codes over $GF(4)$," submitted to *IEEE Trans. Inform. Theory.*

[7] A. R. Calderbank and P. W. Shor, "Good quantum error-correcting codes exist," *Phys. Rev. A*, **54**, pp. 1098–1105 (1996); also LANL e-print quant-ph/9512032.

[8] D. Dieks, "Communication by EPR devices," *Phys. Lett. A*, **92**, p. 271 (1982).

[9] A. Ekert and C. Macchiavello, "Error correction in quantum communication," LANL e-print quant-ph/9602022.

[10] R. Laflamme, C. Miquel, J. P. Paz and W. H. Zurek, "Perfect quantum error correction codes," *Phys. Rev. Lett.* (to appear); also LANL e-print quant-ph/9602019.

[11] F. J. MacWilliams and N. J. A. Sloane, *The Theory of Error-Correcting Codes*, North-Holland, Amsterdam, 1977.

[12] P. W. Shor, "Scheme for reducing decoherence in quantum memory," *Phys. Rev. A*, **52**, p. 2493 (1995).

[13] P. Shor, "Fault-tolerant quantum computation," *Proc. 37th Sympos. Foundations of Computer Science*, 1996, to appear.

[14] A. M. Steane, "Multiple particle interference and quantum error correction," *Proc. Roy. Soc. London A*, submitted; LANL e-print quant-ph/9601029.

[15] A. M. Steane, "Simple quantum error correcting codes," *Phys. Rev. Lett.*, **77**, pp. 793–797 (1996).

[16] W. K. Wooters and W. H. Zurek, "A single quantum cannot be cloned," *Nature*, **299**, p. 802 (1982).

Probabilistic Crisscross Error Correction

RON M. ROTH*

Abstract

The crisscross error model in data arrays is considered, where the corrupted symbols are confined to a prescribed number of rows or columns (or both). Under the additional assumption that the corrupted entries are uniformly distributed over the channel alphabet, a probabilistic coding scheme is presented where the redundancy can get close to one half the redundancy required in minimum-distance decoding of crisscross errors.

1 Introduction

Consider an application where information symbols (such as bits or bytes) are stored in $m \times n$ arrays, with the possibility of some of the symbols recorded erroneously. The error patterns are such that all corrupted symbols are confined to a prescribed number of rows or columns (or both). We refer to such an error model as *crisscross errors*. A crisscross error pattern that is confined to two rows and three columns is shown in Figure 1.

Crisscross errors can be found in various data storage applications; see, for instance, [1], [3], [4], [7], [9], [10], [11], [12]. Such errors may occur in memory chip arrays, where row or column failures occur due to the malfunctioning of row drivers or column amplifiers. Crisscross errors can also be found in helical tapes, where the tracks are recorded in a direction which is (conceptually) perpendicular to the direction of the movement of the tape; misalignment of the reading head causes burst errors to occur along the track (and across the tape), whereas scratches on the tape usually occur along the tape (and across the tracks). Crisscross error-correcting codes can also be applied in linear magnetic tapes, where the tracks are written along the direction of the movement of the tape and, therefore, scratches cause bursts to occur along the tracks; still, the information and check symbols are usually recorded across the tracks. Computation of check symbols is equivalent to decoding of erasures at the check locations, and in this case these erasures are perpendicular to the erroneous tracks.

Crisscross errors can be analyzed through the following cover metric. A *cover* of an $m \times n$ array Γ over a field F is a set of rows or columns that contain all the nonzero entries in Γ. The *cover weight* of Γ is the size of the smallest cover of Γ. The *cover*

*Hewlett-Packard Laboratories, 1501 Page Mill Road, Palo Alto, CA 94304, USA. On sabbatical leave from the Computer Science Department, Technion, Haifa 32000, Israel. This work was done at Hewlett-Packard Israel Science Center, Haifa, Israel. e-mail: ronny@cs.technion.ac.il.

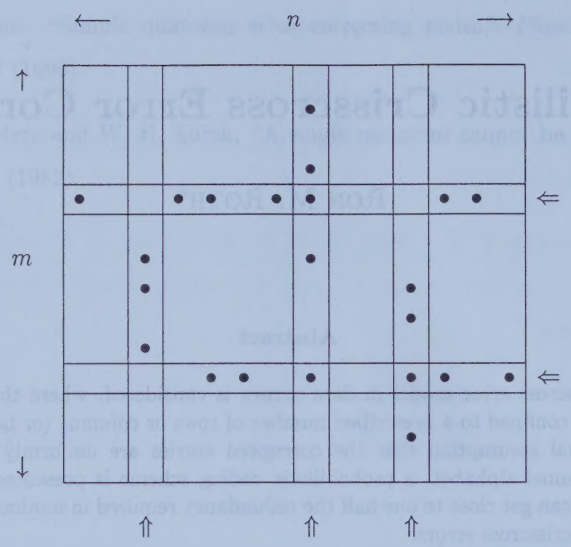


Figure 1: Typical crisscross error pattern.

distance between two $m \times n$ arrays over F is the cover weight of their difference. An $[m \times n, k, d_{\text{cov}}]$ array code over F is a k-dimensional linear subspace $\mathcal{C}$ of the vector space of all $m \times n$ matrices over F such that d_{cov} is the smallest cover distance between any two distinct elements of $\mathcal{C}$ or, equivalently, the smallest cover weight of any nonzero element of $\mathcal{C}$. The parameter d_{cov} is referred to as the *minimum cover distance* of $\mathcal{C}$ and the term $mn-k$ stands for the *redundancy* of $\mathcal{C}$.

The Singleton bound on the minimum cover distance states that the minimum cover distance and the redundancy of any $[m \times n, k, d_{\text{cov}}]$ array code over a field F satisfy the relation

$$mn - k \;\geq\; (d_{\text{cov}} - 1)\, n \;, \tag{1}$$

where we assume that $m \leq n$ (see [6] and [12]).

Let Γ be the "transmitted" array and $\Gamma + E$ be the "received" array, where E is the error array. The number of crisscross errors is bounded from below by the cover weight of E. Since cover distance is a metric, then by using an $[m \times n, k, d_{\text{cov}}]$ array code, we can recover any pattern of up to $(d_{\text{cov}}-1)/2$ crisscross errors. On the other hand, if we wish to be able to recover *any* pattern of up to t crisscross errors, then we *must* use an array code with minimum cover distance which is at least $2t+1$. The Singleton bound on the minimum cover distance implies that the number of redundancy symbols must be at least $2tn$, namely, at least twice the maximum number of erroneous symbols that need to be corrected.

In [6] and [12], it was shown how crisscross errors can be handled by applying array codes for the rank metric. A μ-$[m \times n, k]$ array code $\mathcal{C}$ over a field F is a k-dimensional linear subspace of the vector space of all $m \times n$ matrices over F such that μ is the smallest rank of any nonzero matrix in $\mathcal{C}$. The parameter μ is referred to as the *minimum rank* of $\mathcal{C}$.

The Singleton bound on the minimum rank takes the form

$$mn - k \;\geq\; (\mu - 1)\, n\,, \tag{2}$$

where we assume that $m \leq n$. This bound was stated by Delsarte in [2]; see also Gabidulin [5] and Roth [12]. Furthermore, those references contain a construction of μ-$[n \times n, k]$ array codes over the field $\mathsf{F}_q = GF(q)$ that attains this bound for every $\mu \leq n$. We describe next this optimal construction, which we denote by $\mathcal{C}(n \times n, t; q)$, where $\mu = t{+}1$. Let $\boldsymbol{\alpha} = [\,\alpha_i\,]_{i=1}^{n}$ be a row vector over $\mathsf{F}_{q^n} = GF(q^n)$ and $\boldsymbol{\omega} = [\,\omega_j\,]_{j=1}^{n}$ be a column vector over F_{q^n}, each vector having entries that are linearly independent over F_q. The array code $\mathcal{C}(n \times n, t; q)$ consists of all $n \times n$ matrices $\Gamma = [\,\Gamma_{i,j}\,]_{i,j=1}^{n}$ over F_q such that

$$\sum_{i,j=1}^{n} \Gamma_{i,j}\, \alpha_i^{q^\ell} \omega_j \;=\; 0\,, \qquad 0 \leq \ell < t\,. \tag{3}$$

Two efficient decoding algorithms for $\mathcal{C}(n \times n, t; q)$ are presented in [5] and [12] for recovering any error pattern of rank $\leq t/2$. The construction $\mathcal{C}(n \times n, t; q)$ can be generalized to obtain optimal $(t{+}1)$-$[m \times n, k]$ array codes by means of code shortening. Namely, to form a $(t{+}1)$-$[m \times n, (m{-}t)n]$ array code for $m \leq n$, we take the $m \times n$ upper blocks of all the elements Γ in a $(t{+}1)$-$[n \times n, (n{-}t)n]$ array code such that the last $n{-}m$ rows in each Γ are zero.

The application of μ-$[m \times n, k]$ array codes to crisscross error correction is based upon the observation that matrix rank is a metric and that the cover weight of an array is bounded from below by its rank. By using the elements of a μ-$[m \times n, k]$ array code for transmission (or recording), we can recover any error array of rank $\leq (\mu{-}1)/2$ and, therefore, we can correct any pattern of up to $(\mu{-}1)/2$ crisscross errors. Thus, every μ-$[m \times n, k]$ array code is also an $[m \times n, k, \mu]$ array code. The array codes defined by (3) are optimal with respect to the bound (2) and, as such, they are optimal with respect to (1).

Still, such an optimality criterion is based upon a worst-case decoding strategy where we are interested in being able to decode *any* pattern of up to t crisscross errors, thus requiring to have at least $2tn$ redundancy symbols. The purpose of this work is to show that, by assuming a uniform distribution on the error values in each error location, and by allowing an acceptable probability of miscorrection, significant savings in the redundancy can be obtained. More specifically, we assume that the noisy channel acts on the transmitted array Γ over $\mathsf{F}_q = GF(q)$ as follows:

(P1) The noisy channel selects a set X_r of row indexes and a set X_c of column indexes such that $|\mathsf{X}_\mathrm{r}| + |\mathsf{X}_\mathrm{c}| \leq t$. Note that no assumption is made on the selection of X_r and X_c other than limiting the sum of their sizes to be at most t. In particular, we do not assume any a priori probabilistic behavior on such a selection.

(P2) The channel marks entries in Γ within the rows and columns that were selected in (P1). Again, except for their confinement to the rows and columns indexed by X_r and X_c, no a priori assumption is made on the location of the entries that are marked.

(P3) Each marked entry in Γ is set to a value which is uniformly distributed over F_q independently of the values chosen for the other marked entries in Γ. (In particular, a marked entry may still maintain the correct value with probability $1/q$.)

This probabilistic model of errors seems to approximate rather well the situation in reality, where crisscross errors are caused mainly by bursts. Bursts tend to overwrite the original data in Γ independently of that data. In some cases, however, the retrieved bursty stream, which appears as rows or columns in $\Gamma + E$, might have a relatively small number of typical patterns (e.g., tendency of the patterns to contain runs of the same symbol). In such cases, we make the array Γ appear random by the use of pseudo-random generators (i.e., scramblers), thus forcing the error values in the error array E to look random.

2 Code construction

In this section, we describe a construction of an $[n \times n, k]$ array code $\mathsf{C}(n \times n, t, \mathsf{p}; q)$ over F_q that can correct patterns of up to t crisscross errors with a decoding failure probability which is bounded from above by p. By this probability of failure we mean that for any selection of marked entries according to conditions (P1) and (P2), the probability that the values that were chosen according to (P3) resulted in an uncorrectable error array is at most p. For a wide range of values of t, the miscorrection probability allows the redundancy of these codes to get close to tn, namely one half the redundancy of $\mathcal{C}(n \times n, 2t; q)$. Note that the model allows to have up to tn erroneous symbols in the worst case. Therefore, we must have at least as many as tn redundancy symbols when $\mathsf{p} < 1 - (1/q)$. For the sake of simplicity we deal here with constructions of $[m \times n, k]$ array codes where $m = n$. The general case can be handled by code shortening.

Let C_1 and C_2 be two conventional linear codes of length n, redundancy r, and minimum Hamming distance d over F_q, with $r \times n$ parity-check matrices H_1 and H_2, respectively. The specific value of d (and r) will be set later on.

We define $\mathsf{C}(n \times n, t, \mathsf{p}; q)$ as an $[n \times n, k]$ array code over F_q consisting of all $n \times n$ arrays Γ such that

$$\Gamma \in \mathcal{C}(n \times n, t; q) \qquad \text{and} \qquad \hat{\Gamma} = H_1 \Gamma H_2^T \in \mathcal{C}(r \times r, 2t; q) \, .$$

The redundancy $n^2 - k$ of $\mathsf{C}(n \times n, t, \mathsf{p}; q)$ is bounded from above by $tn + 2tr$.

We next sketch the analysis of the correction capability of $\mathsf{C}(n \times n, t, \mathsf{p}; q)$. For a matrix A over F, we denote by $\mathsf{span}_{\mathsf{c}}(A)$ the linear space over F which is spanned by the columns of A. Similarly, we denote by $\mathsf{span}_{\mathsf{r}}(A)$ the linear space which is spanned by the rows of A. The rank of A will be denoted by $\mathsf{rank}(A)$. Clearly, $\mathsf{rank}(A) = \dim \mathsf{span}_{\mathsf{c}}(A) = \dim \mathsf{span}_{\mathsf{r}}(A)$.

Suppose that $\Gamma \in \mathsf{C}(n \times n, t, \mathsf{p}; q)$ is the transmitted array and that $\Gamma + E$ is the received array where E is an $n \times n$ error array over F_q which was generated by the channel according to conditions (P1)–(P3). Let $\rho = \mathsf{rank}(E)$. We have $\rho \leq t$ and we can write

$$E = UAD \, , \tag{4}$$

where U is an $n \times \rho$ matrix whose columns form a basis of $\mathsf{span}_{\mathsf{c}}(E)$, D is a $\rho \times n$ matrix whose rows form a basis of $\mathsf{span}_{\mathsf{r}}(E)$, and A is a $\rho \times \rho$ nonsingular matrix. Note that the decomposition (4) of E is not unique since U and D could be any bases of $\mathsf{span}_{\mathsf{c}}(E)$ and

$\mathsf{span}_r(E)$, respectively, and for any choice of bases we would have a unique matrix A so that (4) holds.

Let $\hat{E}$ denote the $r \times r$ matrix $H_1 E H_2^T$. Clearly, $\mathsf{rank}(\hat{E}) \leq \mathsf{rank}(E) = \rho \leq t$. Using the decoding algorithm for $\mathcal{C}(r \times r, 2t; q)$ [5],[12] we can recover the array $\hat{E}$. It remains to be shown how the parameters can be tuned so that, with probability of success $\geq 1-\mathsf{p}$, we can recover E out of $\hat{E}$ and the syndrome values of E (which are computed for $\mathcal{C}(n \times n, t; q)$), given our assumption (P3) on the distribution of the entries of E.

We outline here only a skeleton of the proof. Denote by E_c^- the $n \times (n - |\mathsf{X}_c|)$ submatrix of E consisting of columns which are indexed by $\{1, 2, \ldots, n\} - \mathsf{X}_c$. Similarly, the notation E_r^- will stand for the submatrix of E consisting of rows which are indexed by $\{1, 2, \ldots, n\} - \mathsf{X}_r$. The proof consists of the following two parts:

1. Showing that when we choose the linear codes C_1 and C_2 so that

$$d = 3t + \lceil \log_q(2(t+1)/\mathsf{p}) \rceil ,$$

then, with probability at least $1 - \mathsf{p}$, we can find an $n \times \sigma$ matrix U_1 such that

$$\mathsf{span}_c(E_c^-) \subseteq \mathsf{span}_c(U_1) \subseteq \mathsf{span}_c(E) . \tag{5}$$

Similarly, we find a $\tau \times n$ matrix D_1 such that

$$\mathsf{span}_r(E_r^-) \subseteq \mathsf{span}_r(D_1) \subseteq \mathsf{span}_r(E) . \tag{6}$$

2. Showing that, once $\hat{E}$, U_1, and D_1 are determined, the error array E is completely specified.

If C_1 and C_2 are taken to be (extended) BCH codes over F_q, then the value of r will not exceed $(d-1)\lceil \log_q(n-1) \rceil$. Hence, for every real $\epsilon > 0$ we can attain redundancy less than $tn(1+\epsilon)$, provided that $t \leq \delta n / \log_q n$ and $\mathsf{p} = 2^{-\delta n / \log_2 n}$ for some real $\delta > 0$ that depends on ϵ. Furthermore, if C_1 and C_2 are Justesen codes [8, Ch. 10], then for every $\epsilon' > 0$ there is $\delta' > 0$ such that we can have both $r \leq \epsilon' n$ and $d \geq \delta' n$. Using such codes C_1 and C_2, we can attain redundancy less than $tn(1+\epsilon)$ whenever $t \leq \delta n$ and $\mathsf{p} = q^{-\delta n}$ for some $\delta = \delta(\epsilon) > 0$.

3 Decoding

The decoding algorithm consists of the following stages:

Step 1 Decoding the matrix $\hat{E} = H_1 E H_2^T$ from the syndrome values of $\hat{E}$ which are computed for the code $\mathcal{C}(r \times r; 2t; q)$.

Step 2 Computing matrices U_1 and D_1 such that the set inclusions (5) and (6) hold.

Step 3 Computing the matrix A out of the decomposition of $\hat{E}$ into the product $(H_1 U)\, A\, (D H_2^T)$, and recovering E out of U_1, D_1, and A.

Step 1 can be carried out by one of the decoding algorithms for $\mathcal{C}(r \times r; 2t; q)$ [5], [12].

In order to implement Step 2, we can proceed as follows. Denote by $\mathcal{S}(n,t)$ the set of all column vectors $\mathbf{u} \in \mathsf{F}_q^n$ with Hamming weight $\leq t$. Start by picking a column vector $\mathbf{u}_1 \in \mathcal{S}(n,t)$ such that $H_1\mathbf{u}_1 \in \mathsf{span}_c(\hat{E})$. Iterate this by picking column vectors $\mathbf{u}_\ell \in \mathcal{S}(n,t) - \mathsf{span}_c(\mathbf{u}_1\mathbf{u}_2 \ldots \mathbf{u}_{\ell-1})$ such that $H_1\mathbf{u}_\ell \in \mathsf{span}_c(\hat{E})$. Continue in this manner until no more such vectors can be found. The vectors $\mathbf{u}_\ell$ will form the columns of the $n \times \sigma$ matrix U_1. At this point, we do not know yet of an algorithm better than searching over the elements of $\mathcal{S}(n,t)$ (say, according to increasing values of Hamming weight), to find vectors $\mathbf{u}$ such that $H_1\mathbf{u} \in \mathsf{span}_c(\hat{E})$ (and similarly for the rows). Nevertheless, when the crisscross errors are not too short, then there is high probability to have bases of $\mathsf{span}_c(E_c^-)$ and $\mathsf{span}_r(E_r^-)$ that consist wholly of unit vectors. More specifically, it can be shown that if the channel marks in (P2) at least η entries in each one of the rows and columns that were selected in (P1), then the two mentioned bases will consist wholly of unit vectors with probability greater than $1 - 2q^{t-\eta}$. In such a case, the required search is *linear* in n.

We turn now to Step 3. The error array E can be written as

$$E = \begin{bmatrix} U_1 \, U_2 \end{bmatrix} A \begin{bmatrix} D_1 \\ D_2 \end{bmatrix}, \tag{7}$$

where A is a known nonsingular $\rho \times \rho$ matrix, U_1 is a known $n \times \sigma$ matrix with rank σ, and D_1 is a known $\tau \times n$ matrix with rank τ. Our task is to compute the unknown $n \times (\rho - \sigma)$ matrix U_2 and the $(\rho - \tau) \times n$ matrix D_2 in (7), under the assumption that $t + \sigma + \tau \geq 2\rho$. Indeed, this inequality holds in our case since $\rho - \sigma \leq |\mathsf{X}_c|$, $\rho - \tau \leq |\mathsf{X}_r|$, and $|\mathsf{X}_c| + |\mathsf{X}_r| \leq t$.

Let γ denote the rank of the lower-right $(\rho-\sigma) \times (\rho-\tau)$ block of A. By performing elementary operations on rows and columns, we can simplify the matrix A to a permutation matrix and bring the decomposition (7) to the form

$$E = E_{1,1} + E_{1,2} + E_{2,1} + E_{2,2}, \tag{8}$$

such that

$$E_{a,b} = U_{a,b} D_{b,a}, \qquad a, b \in \{1, 2\}, \tag{9}$$

where

$$U_1 = \begin{bmatrix} U_{1,1} \, U_{1,2} \end{bmatrix}, \quad U_2 = \begin{bmatrix} U_{2,1} \, U_{2,2} \end{bmatrix}, \quad D_1 = \begin{bmatrix} D_{1,1} \\ D_{1,2} \end{bmatrix}, \text{ and } D_2 = \begin{bmatrix} D_{2,1} \\ D_{2,2} \end{bmatrix},$$

in which $U_{1,1}$ occupies the first $\sigma+\tau-\rho+\gamma$ columns of U_1, $U_{2,1}$ occupies the first $\rho-\sigma-\gamma$ columns of U_2, $D_{1,1}$ occupies the first $\sigma+\tau-\rho+\gamma$ rows of D_1, and $D_{2,1}$ occupies the first $\rho-\tau-\gamma$ rows of D_2. Permuting the columns of A and the rows of D, we can make A an identity matrix whereas D becomes $[D_{1,1}^T \; D_{2,1}^T \; D_{1,2}^T \; D_{2,2}^T]^T$. By (8) and (9) we indeed have $E = UD$.

Let $\rho_{a,b}$ denote the rank of $E_{a,b}$. We have

$$\rho_{1,1} = \sigma+\tau-\rho+\gamma, \quad \rho_{1,2} = \rho-\tau-\gamma, \quad \rho_{2,1} = \rho-\sigma-\gamma, \quad \text{and} \quad \rho_{2,2} = \gamma,$$

and those ranks sum to ρ. The matrix $E_{1,1}$, as well as the column span of $E_{1,2}$ and the row span of $E_{2,1}$, are known to the decoder.

We now present a decoding algorithm for recovering the matrix $E' = E - E_{1,1}$, whose rank equals $\rho - \rho_{1,1} = 2\rho - \sigma - \tau - \gamma$. In analogy to conventional error-correcting codes, the partial information that we have on the column and row spans of $E_{1,2}$ and $E_{2,1}$ can be viewed as "erasure information," whereas $E_{2,2}$ needs full correction. The ranks of those three unknown matrices satisfy

$$\rho_{1,2} + \rho_{2,1} + 2\rho_{2,2} \;=\; 2\rho - \sigma - \tau \;\leq\; t\,.$$

Pursuing the analogy with conventional error correction, we would expect such an inequality to suffice for recovering the unknown matrices while using the array code $\mathcal{C}(n \times n, t; q)$. Indeed, the results of Section 2 imply that this is indeed the case. The algorithm presented in [12] handles the special case where $\sigma = \tau = 0$ (and $t \geq 2\rho$).

Let E'' denote the sum $E_{1,2} + E_{2,2}$, which is of rank $\rho_{1,2} + \rho_{2,2} = \rho - \tau$. We first show how we can compute a matrix $U_{2,2}$, consisting of γ linearly independent column vectors that form together with the columns of $U_{1,2}$ a basis of $\mathsf{span}_{\mathsf{c}}(E'') = \mathsf{span}_{\mathsf{c}}(E_{1,2}\ E_{2,2})$. Such a basis will then allow us to recover the matrices E'' and $E_{2,1}$ by solving a set of linear equations.

Since the matrix $E_{1,1}$ is known, the decoder can compute the syndrome vector $\mathbf{s} \in \mathsf{F}_{q^n}^t$ of $E' = E - E_{1,1} = [\,e'_{i,j}\,]_{i,j=1}^{n}$ for $\mathcal{C}(n \times n, t; q)$ with respect to the equations (3); namely, the components of $\mathbf{s}$ can be written as

$$s_\ell \;=\; \sum_{i,j=1}^{n} e'_{i,j}\, \alpha_i^{q^\ell} \omega_j\,, \qquad 0 \leq \ell < t\,. \tag{10}$$

Denote by $\mathsf{Y}_{a,b}$ the set of indexes of the $\rho_{a,b}$ columns in U that belong to the submatrix $U_{a,b}$; e.g., if indexes start at 1, then $\mathsf{Y}_{2,1} = \{j : \sigma < j \leq \rho - \gamma\}$. Note that $\mathsf{Y}_{a,b}$ also points at the rows of $D_{b,a}$ in D. We denote by Y the set $\{j : 0 < j \leq \rho\} = \cup_{a,b\in\{1,2\}} \mathsf{Y}_{a,b}$ and by Y' the set $\mathsf{Y} - \mathsf{Y}_{1,1} = \mathsf{Y}_{1,2} \cup \mathsf{Y}_{2,1} \cup \mathsf{Y}_{2,2}$. Clearly, $|\mathsf{Y}'| = \rho - \rho_{1,1} = 2\rho - \sigma - \tau - \gamma$.

Write $U = [\,(U)_{i,k}\,]_{i,k}$ and $D = [\,(D)_{k,j}\,]_{k,j}$ and define the row vector $\boldsymbol{\beta} = [\,\beta_k\,]_{k\in\mathsf{Y}} = \boldsymbol{\alpha} U$ and the column vector $\boldsymbol{\delta} = [\,\delta_k\,]_{k\in\mathsf{Y}} = D\boldsymbol{\omega}$, both in $\mathsf{F}_{q^n}^{\rho}$. By (8), (9), and (10) we have

$$s_\ell = \sum_{k\in\mathsf{Y}'} \beta_k^{q^\ell} \delta_k\,, \qquad 0 \leq \ell < t\,. \tag{11}$$

We now introduce the polynomials $\Phi(x)$, $\Psi(x)$, and $\Lambda(x)$ over F_{q^n}, which are defined by

$$\Phi(x) \;=\; \sum_{m=0}^{\rho_{1,2}} \phi_m x^{q^m} \;=\; \prod_{\mathbf{y}\in\mathsf{span}_{\mathsf{c}}(E_{1,2})} (x - \boldsymbol{\alpha}\mathbf{y})\,,$$

$$\Psi(x) \;=\; \sum_{m=0}^{\rho_{2,1}} \psi_m x^{q^m} \;=\; \prod_{\mathbf{z}\in\mathsf{span}_{\mathsf{r}}(E_{2,1})} (x - \mathbf{z}\boldsymbol{\omega})\,,$$

and

$$\Lambda(x) \;=\; \sum_{m=0}^{\rho-\tau} \lambda_m x^{q^m} \;=\; \prod_{\mathbf{u}\in\mathsf{span}_{\mathsf{c}}(E'')} (x - \boldsymbol{\alpha}\mathbf{u})\,.$$

Those three polynomials are *linearized* polynomials over F_{q^n}, namely, they have the form $\sum_m a_m x^{q^m}$ where $a_m \in \mathsf{F}_{q^n}$. Several properties of linearized polynomials are summarized

in [8, Section 4.9]. In particular, if $a(x) = \sum_m a_m x^{q^m}$ is a linearized polynomial over F_{q^n}, then the mapping $x \mapsto a(x)$ over the domain F_{q^n} is a linear transformation over F_q. Thus, the set of roots of $a(x)$ in F_{q^n} forms a linear vector space over F_q, as this set is a null space of a linear transformation.

Recalling that the values β_k are known for $k \in \mathsf{Y}_{1,2}$, the decoder can compute the polynomial $\Phi(x)$ by solving the following set of $\rho_{1,2} = \rho-\tau-\gamma$ linear equations over F_{q^n} for the coefficients ϕ_m:

$$\Phi(\beta_k) \;=\; \sum_{m=0}^{\rho_{1,2}} \phi_m \beta_k^{q^m} \;=\; 0\,, \qquad k \in \mathsf{Y}_{1,2}\,, \tag{12}$$

with $\phi_{\rho_{1,2}} = 1$. Indeed, the solution for the coefficients ϕ_m is unique since the equations (12) are linearly independent [8, p. 117]. Similarly, the polynomial $\Psi(x)$ can be computed by solving the equations

$$\Psi(\delta_k) \;=\; \sum_{m=0}^{\rho_{2,1}} \psi_m \delta_k^{q^m} \;=\; 0\,, \qquad k \in \mathsf{Y}_{2,1}\,, \tag{13}$$

with $\psi_{\rho_{2,1}} = 1$.

The coefficients of the polynomial $\Lambda(x)$ satisfy the following set of $\rho-\tau-\gamma$ linear equations over F_{q^n}:

$$\Lambda(\beta_k) \;=\; \sum_{m=0}^{\rho-\tau} \lambda_m \beta_k^{q^m} \;=\; 0\,, \qquad k \in \mathsf{Y}_{1,2}\,.$$

Yet, when $\gamma > 0$, we will need additional constraints in order to determine $\Lambda(x)$ uniquely.

Let S be the $(t-\rho+\tau) \times (\rho-\tau+1)$ matrix over F_{q^n} which is defined by

$$S \;=\; \begin{bmatrix} s_0^{q^n} & s_1^{q^n} & \cdots & s_{\rho-\tau}^{q^n} \\ s_1^{q^{n-1}} & s_2^{q^{n-1}} & \cdots & s_{\rho-\tau+1}^{q^{n-1}} \\ s_2^{q^{n-2}} & s_3^{q^{n-2}} & \cdots & s_{\rho-\tau+2}^{q^{n-2}} \\ \vdots & \vdots & \vdots & \vdots \\ s_{t-\rho+\tau-1}^{q^{n-(t-\rho+\tau-1)}} & s_{t-\rho+\tau}^{q^{n-(t-\rho+\tau-1)}} & \cdots & s_{t-1}^{q^{n-(t-\rho+\tau-1)}} \end{bmatrix} \tag{14}$$

and let R denote the $(t+\sigma+\tau-2\rho+\gamma) \times (t-\rho+\tau)$ matrix over F_{q^n} which is given by

$$R = \begin{bmatrix} \psi_g^{q^{n-g}} & \psi_{g-1}^{q^{n-g}} & \cdots & \psi_1^{q^{n-g}} & \psi_0^{q^{n-g}} & 0 & \cdots & & \\ 0 & \psi_g^{q^{n-g-1}} & \psi_{g-1}^{q^{n-g-1}} & \cdots & \psi_1^{q^{n-g-1}} & \psi_0^{q^{n-g-1}} & 0 & \cdots & \\ \cdots & 0 & \psi_g^{q^{n-g-2}} & \psi_{g-1}^{q^{n-g-2}} & \cdots & \psi_1^{q^{n-g-2}} & \psi_0^{q^{n-g-2}} & 0 & \cdots \\ & \cdots & 0 & \ddots & & & \ddots & 0 & \\ & & \cdots & 0 & \psi_g^{q^{n-g'}} & \psi_{g-1}^{q^{n-g'}} & \cdots & \psi_1^{q^{n-g'}} & \psi_0^{q^{n-g'}} \end{bmatrix}, \tag{15}$$

where $g = \rho_{2,1} = \rho-\sigma-\gamma$ and $g' = g+t+\sigma+\tau-2\rho+\gamma-1 = t-\rho+\tau-1$. We denote by R_L the submatrix consisting of the first L rows of R. We state the following three lemmas without proof.

Lemma 1. *Let $\boldsymbol{\lambda} = [\,\lambda_m\,]_{m=0}^{\rho-\tau}$ be the column vector of coefficients of the polynomial $\Lambda(x) = \sum_{m=0}^{\rho-\tau} \lambda_m x^{q^m}$. Then, $RS\boldsymbol{\lambda} = \mathbf{0}$.*

Lemma 2. *Let* $\mathbf{v} = [\,v_m\,]_{m=0}^{\rho-\tau}$ *be the column vector of coefficients of a linearized monic polynomial* $v(x) = \sum_{m=0}^{\rho-\tau} v_m x^{q^m}$ *such that* $v(\beta_k) = 0$ *for* $k \in \mathsf{Y}_{1,2}$ *and* $R_\gamma S\mathbf{v} = \mathbf{0}$. *Then* $v(x) = \Lambda(x)$.

It follows from Lemmas 1 and 2 that for any integer L in the range $\gamma \le L \le t+\sigma+\tau-2\rho+\gamma$, there is a unique monic linearized polynomial $v(x)$ that vanishes at β_k for all $k \in \mathsf{Y}_{1,2}$ and satisfies the set of equations

$$R_L S\mathbf{v} = \mathbf{0}\,.$$

That unique polynomial equals $\Lambda(x)$.

Lemma 3. *Let* $v(x) = \sum_{m=0}^{\rho-\tau} v_m x^{q^m}$ *be a linearized polynomial over* F_{q^n}. *Then* $v(\beta_k) = 0$ *for* $k \in \mathsf{Y}_{1,2}$ *if and only if there is a polynomial* $\Theta(x) = \sum_{m=0}^{\gamma} \theta_m x^{q^m}$ *over* F_{q^n} *such that*

$$v(x) = \Theta(\Phi(x))\,. \tag{16}$$

Let Q denote the $(\gamma+1) \times (\rho-\tau+1)$ matrix over F_{q^n} which is given by

$$Q = \begin{bmatrix} \phi_0 & \phi_1 & \cdots & \phi_{f-1} & \phi_f & 0 & \cdots & & \\ 0 & \phi_0^q & \phi_1^q & \cdots & \phi_{f-1}^q & \phi_f^q & 0 & \cdots & \\ \cdots & 0 & \phi_0^{q^2} & \phi_1^{q^2} & \cdots & \phi_{f-1}^{q^2} & \phi_f^{q^2} & 0 & \cdots \\ & \cdots & 0 & \ddots & & & \ddots & 0 & \\ & & \cdots & 0 & \phi_0^{q^\gamma} & \phi_1^{q^\gamma} & \cdots & \phi_{f-1}^{q^\gamma} & \phi_f^{q^\gamma} \end{bmatrix}, \tag{17}$$

where $f = \rho_{1,2} = \rho-\tau-\gamma$. Then (16) is equivalent to having

$$\mathbf{v} = Q^T\boldsymbol{\theta}\,,$$

where $\mathbf{v} = [\,v_m\,]_{m=0}^{\rho-\tau}$ and $\boldsymbol{\theta} = [\,\theta_m\,]_{m=0}^{\gamma}$.

Lemmas 1, 2, and 3 provide the means by which we can compute the matrix $U_{2,2}$. We first find the coefficients of $\Theta(x)$ by solving the set of equations

$$R_\gamma S\,Q^T\boldsymbol{\theta} = \mathbf{0} \tag{18}$$

for $\boldsymbol{\theta} = [\,\theta_m\,]_{m=0}^{\gamma}$ with $\theta_\gamma = 1$. By those lemmas we have $Q^T\boldsymbol{\theta} = \boldsymbol{\lambda}$. Now, the mapping $x \mapsto \Lambda(x)$ is linear over F_q. Hence, finding a full basis of the linear space (over F_q) of roots of $\Lambda(x)$ in F_{q^n} is equivalent to finding a basis of the null space of an $n \times n$ matrix over F_q which represents the mapping $x \mapsto \Lambda(x)$. We choose $U_{2,2}$ so that the entries of $\boldsymbol{\alpha}U_{2,2}$ extend $\{\beta_k\}_{k\in\mathsf{Y}_{1,2}}$ to span the null space of $x \mapsto \Lambda(x)$. The entries of $\boldsymbol{\alpha}U_{2,2}$ then form the set $\{\beta_k\}_{k\in\mathsf{Y}_{2,2}}$.

At this point, the set of equations (11) have become linear in the remaining unknown variables, which consist of $\{\beta_k\}_{k\in\mathsf{Y}_{2,1}}$ and $\{\delta_k\}_{k\in\mathsf{Y}_{1,2}\cup\mathsf{Y}_{2,2}}$. The number of those variables is $2\rho-\sigma-\tau-\gamma \le t$ and, in view of the results in Section 2, the solution of (11) for these variables yields a unique error array E.

Following is a summary of Step 3 of the outlined decoding algorithm. We assume that the decomposition of E is of the form given by (8)–(9) and that the matrices $U_{1,1}$, $U_{1,2}$, $D_{1,1}$, and $D_{1,2}$ are known. The decoder therefore knows the entries $\{\beta_k\}_{k\in\mathsf{Y}_{1,1}\cup\mathsf{Y}_{1,2}}$ in the row vector $\boldsymbol{\beta} = \boldsymbol{\alpha}U$, as well as the entries $\{\delta_k\}_{k\in\mathsf{Y}_{1,1}\cup\mathsf{Y}_{2,1}}$ in the column vector $\boldsymbol{\delta} = D\boldsymbol{\omega}$.

1. Compute the syndrome vector $\mathbf{s}$ of $E - U_{1,1}D_{1,1}$ with respect to the equations (3).
2. Compute the coefficients of $\Phi(x)$ and $\Psi(x)$ by solving (12) and (13).
3. Compute the matrix $R_\gamma SQ^T$ using (14), (15), and (17).
4. Compute the coefficients of $\Theta(x)$ by solving (18).
5. Find a matrix $U_{2,2}$ so that the entries of $\boldsymbol{\alpha}U_{2,2}$ extend $\{\beta_k\}_{k\in Y_{1,2}}$ to form the null space of the mapping $x \mapsto \Lambda(x) = \Theta(\Phi(x))$ over F_q.
6. Find the remaining unknown components of $\boldsymbol{\beta}$ and $\boldsymbol{\delta}$ by solving the set of (already linear) equations (11).
7. The error array E equals UD, where $\boldsymbol{\beta} = \boldsymbol{\alpha}U$ and $\boldsymbol{\delta} = D\boldsymbol{\omega}$.

References

[1] M. Blaum, R.J. McEliece, *Coding protection for magnetic tapes: a generalization of the Patel-Hong code, IEEE Trans. Inform. Theory,* IT-31 (1985), 690–693.

[2] Ph. Delsarte, *Bilinear forms over a finite field, with applications to coding theory, J. Comb. Th. A,* 25 (1978), 226–241.

[3] S.A. Elkind, D.P. Siewiorek, *Reliability and performance of error-correcting memory and register codes, IEEE Trans. Computers,* C-29 (1980), 920–927.

[4] P.G. Farrell, *A survey of array error control codes, Europ. Trans. Telecomm. Rel. Technol.,* 3 (1992) 441–454.

[5] E.M. Gabidulin, *Theory of codes with maximum rank distance, Probl. Peredach. Inform.,* 21 (1985), 3–16 (in Russian; pp. 1–12 in the English translation).

[6] E.M. Gabidulin, *Optimal array error-correcting codes, Probl. Peredach. Inform.,* 21 (1985), 102–106 (in Russian).

[7] L. Levine, W. Meyers, *Semiconductor memory reliability with error detecting and correcting codes, Computer,* 9 (Oct. 1976), 43–50.

[8] F.J. MacWilliams, N.J.A. Sloane, *The Theory of Error-Correcting Codes,* North-Holland, Amsterdam, 1977.

[9] W.F. Mikhail, R.W. Bartoldus, R.A. Rutledge, *The reliability of memory with single-error correction, IEEE Trans. Computers,* C-31 (1983), 560–564.

[10] A.M. Patel, S.J. Hong, *Optimal rectangular code for high density magnetic tapes, IBM J. Res. Dev.,* 18 (1974), 579–588.

[11] P. Prunsinkiewicz, S. Budkowski, *A double track error-correction code for magnetic tape, IEEE Trans. Computers,* C-25 (1976), 642–645.

[12] R.M. Roth, *Maximum-rank array codes and their application to crisscross error correction, IEEE Trans. Inform. Theory,* IT-37 (1991), 328–336.

Covering lists in maximum likelihood decoding

Ilya Dumer
College of Engineering
University of California, Riverside, CA 92521.

Abstract

Let a q-ary linear code $C(q,n,r)$ of length n and rate r be used over any memoryless channel $(X^n, Y^n, P(y|x))$. For a given output $y \in Y^n$ we consider the list $X(y,N)$ of N most probable input sequences $x \in X^n$, and say that $X(y,N)$ is the covering list if it includes at least one codeword. We state that for some N of exponential size $q^{n(1-r)}$ the sets $X(y,N)$ form the covering lists for all y and virtually all (long) linear codes. This extends the results on coverings by Hamming spheres to the lists of arbitrarily weighted vectors. We then prove that maximum-likelihood (ML) decoding of virtually all linear codes can be accomplished in about $q^{nr(1-r)/(1+r)}$ operations. This gives up to threefold reduction in complexity exponent of ML decoding relative to the known upper bound $q^{n\min(r,1-r)}$. Also, this complexity is below the lower bounds on trellis complexity.

1 Introduction

The main objective of this paper is to provide some insight as to how near-maximum-likelihood (NML) decoding algorithms, recently presented in [6], are linked to explicit ML decoding. In essence, we prove that the former being suboptimal for any code, are yet optimal for virtually all linear codes.

First, let X denote a q-ary input alphabet and Y be an output alphabet, which can be infinite. Also, let $(X^n, Y^n, P(y|x))$ be a channel defined by an input space X^n, output space Y^n, and conditional probability $P(y|x)$ of receiving an output $y \in Y^n$ for a given $x \in X^n$. A q-ary code $C(q,n,r)$ of length n and code rate r is used over the channel. In the sequel, we suppose that all q^{nr} messages are equiprobable. Then for a given output y ML decoding $\psi(y) = c$ retrieves a codeword $c \in C$ with the maximum *a posteriori* probability $P(c|y) \geq P(c'|y), c' \neq c$, and is optimal. For a given y let $X(y,N)$ be the list of N most probable vectors $x \in X^n$ (which give N maximum *a posteriori* probabilities $P(x|y)$ among all q^n vectors).

Now for a given y we consider decoding algorithm $\Psi_N(y)$, which seeks the most probable codeword *only* in the list $X(y,N)$ and fails to decode, whenever $X(y,N)$ includes no codewords. Therefore the decoding error probability P_{Ψ_N} of the algorithm $\Psi_N(y)$ can be greater than the probability P_ψ of ML decoding $\psi(y)$. Yet, we prove in [6] that for the so-called symmetric channels the latter increase is marginal in the sense that for sufficiently large $N = lq^{n(1-r)}$ the inequality

$$P_{\Psi_N} \leq P_\psi(1 + 1/l) \tag{1}$$

holds *for any linear or nonlinear code.* Taking l that slowly grows with n, we can guarantee that asymptotic NML performance of *long* codes is as good as that of ML decoding. As for *any (short) code*, we can take $l = 1$ and bound the decoding error probability of NML decoding by twice the error probability of its ML decoding. In this way, any algorithm $\Psi_N(y)$ is said to provide NML performance, if for long codes its decoding error probability P_{Ψ_N} tends to P_ψ. Also, the NML algorithm $\Psi_N(y)$ from [6] requires exponential order of $q^{nr(1-r)}$ operations for all cyclic codes and virtually all long linear codes. This substantially reduces the known upper bound $q^{n\min(r,1-r)}$ [20] on complexity of ML decoding. It is also important that complexity of NML decoding already falls off below the lower bounds on trellis complexity.

On the other hand, NML decoding is currently developed only for symmetric channels. The latter can be continuous or discrete, additive or nonadditive provided that we use signals of equal energy. The examples include the AWGN channels with Q-PSK signaling, hard decision channels with the Hamming or Lee metric, and many others. However, an important open issue is whether this NML decoding can be generalized for the *nonsymmetric* channels that use signals with distinct energies, as for instance in QAM signaling. Another important issue is whether NML decoding can provide explicit ML performance for general linear codes.

In this paper we give positive answers to both questions for long linear codes. First, we prove that for virtually all linear codes used over any *memoryless* channel, the list $X(y, N)$ of the exponential size $q^{n(1-r)}$ includes at least one codeword for any output y. This means that algorithm $\Psi_N(y)$ provides explicit ML performance for both symmetric and nonsymmetric channels. Second, we design the ML decoding algorithm $\Psi_N(y)$, whose complexity is bounded by the exponential order of $q^{nr(1-r)/(1+r)}$ operations for virtually all linear codes. These results generalize previously known results on NML decoding. Note, however, that the bound (1) can be applied to any (short) code. So an important issue that we leave open, is how the results of this paper can be applied to practically important short codes.

2 Background

Let $p(b|a)$ denote any conditional probability distribution defined over the pairs (b, a) of symbols $b \in Y, a \in X$. Below we consider memoryless channels, in which case $P(y|x) \stackrel{def}{=} \prod_{j=1}^{n} p(y_i|x_i)$ for any pair x, y. Now for a given output y define the "soft decision weight"

$$w_j(a) = -\ln p(a|y_j) \tag{2}$$

of any input symbol $a \in X$ in any position j. We then consider the $(q \times n)$ matrix $W = (w_j(a))$ defined over all input symbols (rows) $a = 0, .., q-1$ and all positions (columns) $j = 1, .., n$. We also define the total "soft decision weight"

$$w(x, W) = \sum_{j=1}^{n} w_j(x_j) = -\ln P(x|y). \tag{3}$$

of any vector $x = (x_1, \ldots, x_n)$. Now any output y and a channel are fully replaced by a matrix W. In particular, the most probable codeword c is the lightest one in terms of the minimum weight $w(c, W)$. We can also regard $X(y, N)$ as the list $X(N, W)$ of N lightest vectors x (which have minimal weights $w(x, W)$ among all q^n vectors). Below

we are mostly interested in whether the lists $X(N, W)$ necessarily include codewords for sufficiently large N.

We first consider *hard decision* decoding in the Hamming metric, in which case $Y = X$ and

$$w_j(a) = \begin{cases} 0, & \text{if } a = y_j, \\ 1, & \text{if } a \neq y_j. \end{cases}$$

In particular, minimum distance (MD) decoding necessarily maps a received vector y onto the closest codeword in the Hamming metric. Also, the set $X(y, N)$ of N "lightest" vectors belongs to the Hamming sphere $V_{D(N)}(y)$ of radius

$$D(N) = \max\{d : \sum_{i=0}^{d-1} (q-1)^i \binom{n}{i} \leq N\} \tag{4}$$

centered at the received vector y. Note that $D(q^{n(1-r)})$ gives the Gilbert distance D for given (q, n, r), and that $D \sim \delta n$ for $n \to \infty$. Here $\delta = H_q^{-1}(1-r)$, and $H_q(z) = -z \log_q z - (1-z) \log_q (1-z) + z \log_q (q-1)$ is the q-ary entropy for $0 < z < 1$.

It is proved by Evseev [10] that correction of only D or fewer errors at most doubles the decoding error probability of MD decoding for *any* linear (q, n, r) code. Another important result by Blinovskii [3] states that virtually all long linear (q, n, r) codes meet the Goblick bound on *covering radius* $\rho = D + O(\ln n)$. In other words, correction of slightly more than D errors necessarily provides full MD decoding for virtually all long linear codes. In turn, bounding the number of errors allows to decrease the asymptotic complexity q^{cn} of MD decoding. In particular, the complexity exponents $c_1 = r(1-r)$ [10], $c_2 = \min(r, H_q(2\delta) - 1 + r)$ [16], $c_3 = (1-r)/2$ [8], $c_4 = [H_2(\delta) - (1-r)H_2(\frac{\delta}{1-r})]/\log_2 q$ from [14], [5], and c_5 [9] are obtained in this way. The corresponding bounds are plotted in Fig. 1.

For soft decision (ML) decoding the situation is different. The major problem is that the set $X(N, W)$ of the lightest vectors substantially depends on a specific matrix W in contrast to MD decoding, for which this set forms the sphere $V_{D(N)}(y)$ for any y. Following [1], [20], [17], and [11], most papers develop ML decoding via trellis realization of a code. In [20] trellis ML decoding was proved to give the complexity exponent $c_0 = \min(r, 1-r)$ for arbitrary memoryless channel and any linear (q, n, r)–code. For the best binary concatenations (meeting the asymptotic GV bound) the complexity exponent $c_2 = \min(r, H_2(2\delta) - 1 + r)$ is known [7]. It is shown in [21] that c_2 is the tight lower bound on trellis complexity of binary codes, provided that the GV bound is tight.

The lower bounds on trellis complexity were derived in [19], [21], and [15]. In general, code design with small trellis complexity has been of great interest (see [4], [12], [13], [18] and many more references therein). Yet, NML decoding [6] gives even lower complexity exponent $c_1 = r(1-r)$ for symmetric channels. In Figure 2 we plot c_1 along with the new bound $c_6 = r(1-r)/(1+r)$ and other bounds on ML decoding complexity. Now the main issue of this paper is whether such a reduction in decoding complexity can also be obtained for ML decoding in arbitrary memoryless channels.

Comparison of the above mentioned results [10] and [3] gives a partial answer for hard decision decoding. Namely, the algorithm Ψ_T with the list of size $T = q^{n(1-r)}$ provides near-MD decoding for *any code* $C(q, n, r)$. On the other hand, the list of size $N = \sum_{i=0}^{\rho} (q-1)^i \binom{n}{i}$ provides full MD decoding for *virtually all* long linear codes. Note

that N and T have *the same exponential* order. In other words, the near-MD algorithm Ψ_N being suboptimal for any code, is yet optimal for virtually all linear codes. More general answer for arbitrary memoryless channel is given below. Basically, we prove that NML decoding with a list of the exponential order $q^{n(1-r)}$ is optimal for virtually all linear codes in any memoryless channel.

3 Summary of results

Preliminary remarks

For a given matrix W let a_j^* be the lightest input in position j and $a^* = (a_1^*, ..., a_n^*) \in X^n$ be the lightest input vector. Without loss of generality, we can consider the $(q \times n)$ matrix V with entries $v_j(a) = w_j(a) - w_j(a_j^*)$ and redefine the weight $w(x, V) = w(x, W) - w(a^*, W)$ of any x. So a^* has null weight and is said to be *the center* of the list $X(N, V)$. Below we consider the set $\{V\}$ of all $(q \times n)$ matrices with nonnegative entries. In particular, Hamming metric corresponds to matrix H such that each column (position) has exactly one zero and $q-1$ ones. Then the list $X(N, H)$ belongs to the sphere $V_{D(N)}(a^*)$. Now we are mostly interested in upper and lower bounds on the number $|X(N, V) \cap C(q, n, r)|$ of codewords in the list $X(N, V)$, as matrices V run through the set $\{V\}$. If for a given code C any set $X(N, V)$ includes at least one codeword, then the algorithms Ψ_N provide ML decoding of code C in *any memoryless channel*. So, our main problem is to define a number N such that all the lists $X(N, V)$ are covering. In this paper, we define N up to the exponential order. More precisely we use the following definitions.

Definition 1 *We say that the function $f(n)$ has exponential order q^{nb} and write $ord_q f = b$, if $\log_q(|f(n)|)/n \to b$, as $n \to \infty$.*

Definition 2 *We say that a code $C(q, n, r)$ is N-good, if for every matrix $V \in \{V\}$ the list $X(N, V)$ of N lightest vectors includes at least one codeword. In other words, $C(q, n, r)$ is N-good if*

$$\min_V |X(N, V) \cap C(q, n, r)| \geq 1.$$

Then our results can be summarized in the following three theorems.

Theorems

Theorem 1 *For virtually all linear codes $C(q, n, r)$ the number of codewords in every list $X(N, V)$ is upper-bounded by the exponential order $q^{n(ord_q N - 1 + r)}$:*

$$ord_q\{\max_V |X(N, V) \cap C(q, n, r)|\} \leq ord_q N - 1 + r. \quad (5)$$

Note that for the Hamming metric the inequality (5) upper-bounds the maximal number of codewords in a sphere $V_{D(N)}(a^*)$ centered at any point $a^* \in X^n$. This bound is well known for virtually all linear codes (see [2] and [3]). Theorem 1 generalizes these results for arbitrary lists $X(N, V)$, whose parameters V run through the continuum set $[0, \infty)^{qn}$ of qn nonnegative numbers, in contrast to the Hamming spheres, which are fully defined by q^n centers.

Theorem 2 *For some number N of exponential order $ord_q N = 1 - r$, virtually all linear (q, n, r)-codes are N-good:*

$$\exists\, N : ord_q N = 1 - r,\ \min_V |X(N, V) \cap C(q, n, r)| \geq 1.$$

This theorem states that all the lists $X(N, V)$ for a properly chosen N of exponential size $q^{n(1-r)}$ do include codewords. The proof is based on Theorem 1 and then uses the same arguments as those in [3].

Corollary 1 *Their exists a number N of exponential order $ord_q N = 1 - r$, such that the algorithms Ψ_N provide ML decoding of virtually all long linear codes $C(q, n, r)$ in any memoryless channel.*

The next theorem addresses the problem of designing ML decoding algorithms with reduced complexity.

Theorem 3 *ML decoding of virtually all linear $(q, n, r)-$codes can be executed with complexity of order $q^{nr(1-r)/(1+r)}$ in any memoryless channel.*

Discussion and lemmas

First, we give some comments to the proof of Theorem 1. For a given V let the vectors from the list $X(N, V) = \{x^{(1)}, ..., x^{(N)}\}$ be presorted in order of increasing weights $w(x^{(i)}, V)$. Let $\rho = w(x^{(N)}, V)$ be the maximum weight in the set $X(N, V)$. We can also regard N as a function of ρ and denote the set $X(N, V)$ as $X_\rho(V) = \{x \in X^n | w(x, V) \leq \rho\}$. For a given matrix V we say that a code C is bad [1], if there are too many codewords in $X(N, V) : ord_q |X(N, V) \cap C(q, n, r)| > ord_q N - 1 + r$. Then we refine the technique of [2] and prove that for any V the fraction of bad codes declines as $\exp\{-\alpha n^2\}$ with $\alpha > 0$. In the Hamming metric, this allows to prove (5) for virtually all codes, since the number q^n of different spheres increases much slower than $\exp\{n^2\}$ and the overall fraction of bad codes, taken over all spheres, still declines as $\exp\{-\beta n^2\}$ for some $\beta > 0$. By contrast, in general case this technique cannot be used due to the huge continuous set of different matrices V. The idea of the proof of Theorem 1 is to enclose these lists into a finite number of larger *superlists* $\widehat{X_\rho}(V)$, which yet have the same exponential size $ord_q N$. Below we show that all the lists $X_\rho(V)$ can be enclosed in $\exp\{\gamma q n \ln n\}$ superlists. Then the above arguments can be directly applied to the superlists, since their number $\exp\{\gamma q n \ln n\}$ still grows much slower than the fraction of bad codes declines. These superlists are constructed in two steps.

First, we "rescale" V into the matrix λV using any number $\lambda > 0$. Obviously, $X(N, V) = X(N, \lambda V)$. Yet, we prove that for some λ:

1. All the vectors in $X(N, \lambda V)$ include only the symbols of weight n or less.
2. There exist sufficiently many vectors in $X(N, \lambda V)$, each of which includes at least one symbol of weight 1 or more.

Here "sufficiently many" means that the number L of these vectors has the same exponential order $ord_q N$ as N. Now we can consider only the rescaled matrices V . So we have the following lemma.

[1] More precisely, we consider a sequence of codes of increasing length n

Lemma 1 *There exist sufficiently many vectors in the list $X(N,V)$, which have at least one symbol with the weight within the interval $[1,n]$.*

Then we consider the exponential size $N = N(\rho, V)$ of the list $X_\rho(V)$ as a function of the maximal weight ρ in the list.

Lemma 2 *For any V the exponential size $ord_q N(\rho, V)$ is a continuous function of the maximal weight ρ.*

Now consider the discrete set of numbers $I = \{i/n^2, i = 0, 1, ..., n^3\}$. For a given V let $\widehat{V}$ be the $(q \times n)$ matrix, whose entries are obtained by rounding down the entries of V to the closest number from the set I. Since each of qn entries in V can take any of n^3 values, there exist $n^{3qn} = \exp\{\gamma qn \ln n\}$ matrices $\widehat{V}$. Now for a given list $X_\rho(V)$ define its superlist

$$\widehat{X}_\rho(V) = \{x \in X^n | w(x, \widehat{V}) \leq \rho\}.$$

Obviously, $X_\rho(V) \subseteq \widehat{X}_\rho(V)$. On the other hand, the following holds.

Lemma 3 *The superlist $\widehat{X}_\rho(V)$ has the same exponential size $ord_q N$ as that of $X_\rho(V)$: $ord_q |\widehat{X}_\rho(V)| = ord_q N$.*

Now Theorem 1 follows from Lemma 3 and the above discussion. In turn, Theorem 2 follows from Theorem 1 by applying the technique of [3]. In the next section we briefly describe the algorithm that justifies Theorem 3.

4 Algorithm

The following algorithm combines the techniques of [6] and [8] for linear codes $C[q,n,k]$ of any dimension $k = nr$. In essence, we take the list of exponential size $ord_q N = 1 - r$, satisfying Theorem 2, and provide ML decoding Ψ_N. Decoding is executed n times. Let $L = L(j,s)$ be the subblock, which starts in position $j = 0, ..., n-1$ and includes $s = \lceil 2kn/(n+k) \rceil$ cyclically consecutive positions $\{j, j+1 (\mathrm{mod}\, n), ..., j+s-1 (\mathrm{mod}\, n)\}$. Obviously, $s > k$. For simplicity, let also s be an even number: $s = 2l$, $l = \lceil kn/(n+k) \rceil$. In each trial j we consider decoding on the subblock $L(j,s)$. Truncation of the original code $C[q,n,k]$ to the subblock $L(j,s)$ gives the subcode $C^{(j)}[q,s,k^{(j)}]$ of some dimension $k^{(j)}$. In what follows, we consider codes $C[q,n,k]$ such that $k^{(j)} = k$ for all j. In particular, this fact can be readily verified for all cyclic codes and virtually all linear codes. Then each code $C^{(j)}$ can be represented by the $(s-k) \times s$ parity check matrix $H^{(j)}$. For any subset of positions L and any matrix F with columns f_i, $i \in L$, let $F_I = (f_i, i \in I)$ be the submatrix defined over the subset $I \subset L$. Below we first give a heuristic description of the algorithm, when $s|n$ and q is even.

Algorithm.

1. For the given output $y = (y_0, \ldots, y_{n-1})$ define the "soft-decision weights" $w_j(a)$ of each input symbol $a \in X$ in each position $j = 0, \ldots, n-1$ as in (2). Form the corresponding $(q \times n)$ matrix W.

2. In the trial $j = 0, ..., n-1$ we split the subblock $L(j,s)$ into two halves $I = L(j,l)$ and $J = L(j+l,l)$ of l cyclically consecutive positions. We take $M = N^{l/n}$, and

construct the sets $X_I = X(2M, W_I)$ and $X_J = X(2M, W_J)$ of $2M$ lightest subvectors on the sets I and J. For each subvector $x_I \in X_I$ and $x_J \in X_J$ we also calculate their syndromes $h(x_I) = H_I^{(j)} \times (x_I)^T$ and $h(x_J) = H_J^{(j)} \times (x_J)^T$. Below we consider the syndromes $h(x_I)$ and $h(x_J)$ as natural numbers, that can take values from 0 to $q^{s-k} - 1$.

3. Combine the sets X_I and X_J into the list $Z = X_I \cup X_J$ of size $4M$ and sort the elements z of this list by the increasing order of their syndromes $h(z)$. Take all the (neighbouring) pairs x_I and x_J from Z with equal syndromes. Since $h(x_I) + h(x_J) = 0$ for these pairs, the corresponding vector (x_I, x_J) of length s is a codeword of code $C^{(j)}$. Each codeword (x_I, x_J) is then encoded into a codeword $c \in C$, and the lightest codeword c is chosen.

4. Steps 2 and 3 of the trial j are repeated s times. In each *subtrial* $i = j, ..., j+s-1$ we split the subblock $L(j, s)$ into two halves $I = L(i, l)$ and $J = L(i+l, l)$, and proceed as above. In other words, we use s cyclic shifts of the original halves I and J. The lightest codeword c is chosen through all s subtrials.

5. Step 4 is repeated n times for each trial $j = 0, ..., n-1$, and the lightest codeword is chosen.

In order to justify Theorem 3 we first need to prove that the above algorithm does provide decoding Ψ_N. Second, we need to upper-bound the complexity of this algorithm. The first part follows from the lemmas below.

Lemma 4 *[6]. For every vector $x \in X(N, W)$ there exists at least one subblock $L = L(j, s)$ such that subvector x_L belongs to the sublist $X_L = X(N^{s/n}, W_L)$ of only $N^{s/n}$ lightest subvectors on $L(j, s)$.*

This lemma guarantees that we can consider only $N^{s/n} = M^2$ lightest subvectors x_L for at least one subblock $L(j, s)$ in Step 5.

Lemma 5 *For any subvector $x_L \in X(M^2, W_L)$ there exists at least one partition of the subblock $L(j, s)$ into two halves $I = L(i, l)$ and $J = L(i+l, l)$ such that both halves x_I and x_J belong to the sets of $2M$ lightest subvectors: $x_I \in X(2M, W_I)$ and $x_J \in X(2M, W_J)$.*

This lemma ensures that in Step 4 we find the partition that places both subblocks x_I and x_J into the sublists of $2M$ lightest subvectors. Then the list of codewords $\{(x_I, x_J)\} = C^{(j)} \cap X(M^2, W_L)$ is necessarily constructed through the steps 2-4. Encoding procedure ensures that we obtain any codeword from the list $C \cap X(N, W)$. So, we obtain ML decoding Ψ_N indeed. The complexity of the above algorithm is justified by the following lemmas.

Lemma 6 *[6]. For any M and any subblock L the set $X_L = X(M, W_L)$ of M lightest subvectors can be constructed and presorted in $O(Mn \log_2 M)$ operations.*

This lemma shows that the complexity exponent of Step 2 has the required exponential order $c = r(1 - r)/(1 + r)$. Now we need to upper-bound the size of the list $C^{(j)} \cap X(M^2, W_L)$, since each vector from this list is obtained and encoded in Step 3.

Lemma 7 *For virtually all linear codes C any list $C^{(j)} \cap X(M^2, W_L)$ has the exponential order at most M^2/q^{s-k} :*

$$ord_q\{\max_{j,W} |C^{(j)} \cap X(M^2, W_L)|\} \leq M^2/q^{s-k}$$

This lemma shows that for the given $M^2 = N^{s/n}$ the above list also has exponential size q^{nc}. Note also that changing s would increase complexity either in Step 2 or in Step 3. So, the above choice $s = \lceil 2kn/(n+k) \rceil$ is optimal. Now Theorem 3 follows from the fact that Steps 4 and 5 run only ns trials, and the complexity exponent is defined only by Steps 2 and 3.

Remarks. For odd q the only difference is that in Step 2 we use the "negative" syndromes $\widehat{h}(x_J) = -x_J H_J^T$ for the second half. Then the matching equality $h(x_I) = \widehat{h}(x_J)$ once again gives the codewords (x_I, x_J). For the general case $s \nmid n$, the so-called [6] well decomposable vectors x_I, x_J are designed in Step 2 instead of the lightest vectors.

References

[1] L.R. Bahl, J. Cocke, F. Jelinek, and J. Raviv, "Optimal decoding of linear codes for minimizing symbol error rate," *IEEE Trans. Inform. Theory,* vol. 20, pp. 284-287, 1974.

[2] A.M. Barg and I.I. Dumer, "The algorithm of cascade decoding with incomplete inspection of code vectors", *Probl. Peredachi Inform.*, vol. 22, no. 1, pp. 3-10, 1986 (in Russian).

[3] V.M. Blinovskii, "Lower asymptotic bound on the number of linear code words in a sphere of given radius in F_q^n," *Probl. Peredachi Inform.*, vol. 23, no. 2, pp. 50-53, 1987 (in Russian).

[4] A.R. Calderbank and N.J.A. Sloane, "New trellis codes based on lattices and cosets," *IEEE Trans. Inform. Theory*, vol. 33, pp. 177-195, 1987.

[5] J.T. Coffey and R.M.F. Goodman, "The complexity of information set decoding", *IEEE Trans. Inform. Theory,* vol. 36, pp. 1031-1037, 1990.

[6] I. Dumer, "Suboptimal decoding of linear codes. Partition technique," *IEEE Trans. Inform. Theory,* vol. 42, Nov. 1996 (to appear).

[7] I.I. Dumer, "On the complexity of maximum likelihood decoding of the best concatenated codes", In: *Proc. 8 All-Union Conf. on Coding Theory, vol. 2*, Moscow-Kuibyshev, p. 66-69, 1981 (in Russian).

[8] I.I. Dumer, "Two algorithms for linear codes decoding", *Probl. Peredachi Inform.*, vol. 25, no. 1, pp. 24-32, 1989 (in Russian).

[9] I.I. Dumer, "On minimum distance decoding of linear codes", *In: Proc. Fifth Intern. Workshop on Information Theory "Convolutional Codes; Multiuser Communication"*, Moscow, pp. 50-52, Jan. 1991.

[10] G.S. Evseev, "On the complexity of decoding linear codes", *Probl. Peredachi Inform.*, vol. 19, no. 1, pp. 3-8, 1983 (in Russian).

[11] G.D. Forney, Jr., "Coset codes II: Binary lattices and related codes", *IEEE Trans. Inform. Theory*, vol. 34, pp. 1152-1187, 1988.

[12] G.D. Forney, Jr., "Dimension/length profiles and trellis complexity of linear block codes," *IEEE Trans. Inform. Theory*, vol. 40, pp. 1741-1752, 1992.

[13] G.D. Forney, Jr. and M.D. Trott, "The dynamics of group codes: state spaces, trellis diagrams and canonical encoders," *IEEE Trans. Inform. Theory*, vol. 39, pp. 1491-1513, 1993.

[14] E.A. Krouk, "A bound on the decoding complexity of linear block codes", *Probl. Peredachi Inform.*, vol. 25, no. 3, pp. 103-106, 1989 (in Russian).

[15] A. Lafourcade and A. Vardy, "Lower bounds on trellis complexity of block codes," *IEEE Trans. Inform. Theory*, vol. 41, pp. 1938-1952, 1995.

[16] L. Levitin and C. Hartmann, "A new approach to the general minimum distance decoding problem: The zero-neighbors algorithm", *IEEE Trans. Inform. Theory*, vol. 31, pp. 378-384, 1985.

[17] J.L. Massey, "Foundations and methods of channel coding", *Proc. Int. Conf. Inform. Theory and Systems*, NTG-Fachberichte vol. 65, pp. 148-157, 1978.

[18] R.J. McEliece, "On the BCJR trellis for block codes," *IEEE Trans. Inform. Theory*, vol. 42, pp. 1072-1092, 1996.

[19] D.J. Muder, "Minimal trellises for block codes", *IEEE Trans. Inform. Theory*, vol. 34, pp. 1049-1053, 1988.

[20] J.K. Wolf, "Efficient maximum likelihood decoding of linear codes using a trellis", *IEEE Trans. Inform. Theory*, vol. 24, pp. 76-80, 1978.

[21] V.V. Zyablov and V.R. Sidorenko, "Bounds on complexity of trellis decoding of linear codes," *Probl. Peredachi Inform.*, vol. 29, no. 3, pp. 3-9, 1993 (in Russian).

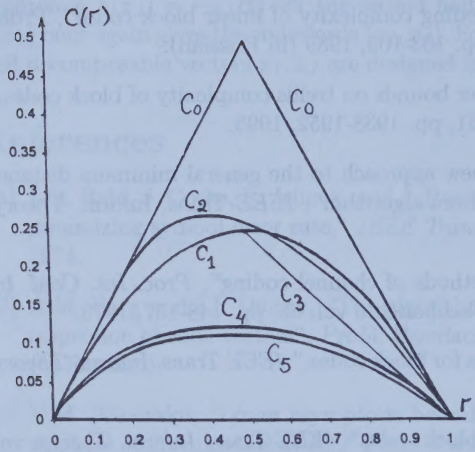


Fig. 1- MD decoding

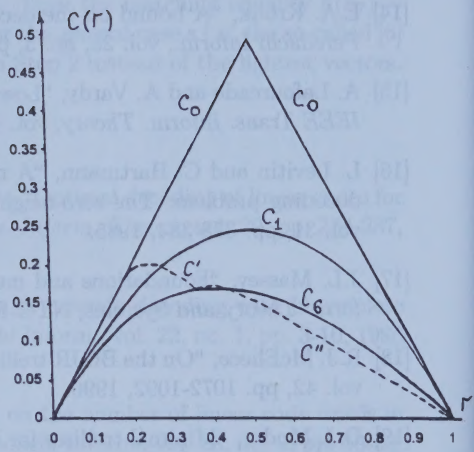


Fig. 2 - ML decoding

Complexity exponents versus code rate r for binary codes

$\delta = H_2^{-1}(1-r)$	Relative GV distance
$c_0 = \min(r, 1-r)$	[Wolf 78]
$c_1 = r(1-r)$	[Evseev 83: MD decoding] [Dumer 93: ML decoding]
$c_2 = \min(r, H_2(2\delta) - 1 + r)$	[Hartmann-Levitin 85]
$c_3 = (1-r)/2$	[Dumer 89]
$c_4 = (1-r)[1 - H_2(\frac{\delta}{1-r})]$	[Krouk 89] [Coffey-Goodman 90]
c_5	[Dumer 91]
$c_6 = r(1-r)/(1+r)$	[This paper]
c' lower bounds	[Zyablov-Sidorenko 93]
c''	[Lafourcade-Vardy 95]

Compact Description of Golay Sequences and Their Extensions

Rüdiger Urbanke[1] and A.S. Krishnakumar[2]

[1] Lucent Technologies, Bell Labs Innovations, Room 2C-254
600 Mountain Avenue, P.O. Box 636
Murray Hill, NJ 07974

[2] Lucent Technologies, Bell Labs Innovations, Room 2E-1
Zadelstede 1-10, Nieuwegein
3431 JZ Netherlands

September 30, 1996

Abstract

We introduce new techniques to derive compact descriptions of Golay sequences and their extensions. In particular we show how a fundamental notion of commutative algebra, namely the notion of Gröbner bases, can be successfully employed to give compact description of Golay sequences. We then concentrate on the practically important case of Golay sequences of length 2^n. For those sequences we give a recursive definition of a subset of size $2^n n!$, which we conjecture to coincide with the set of Golay sequences.

Key words: Golay sequences, polyphase sequences, multitone, Gröbner basis

1 Introduction

In 1951, and more detailed in 1961, Golay investigated special pairs of ± 1 sequences, which he called *complementary* pairs, for their use in spectrometry, [1, 2], see also [3]. These are pairs of sequences of length N such that their aperiodic autocorrelations add up to an impulse of magnitude $2N$. More precisely, let $\boldsymbol{x}, \boldsymbol{y} \in \{\pm 1\}^N$. We say that $\boldsymbol{x}$ and $\boldsymbol{y}$ are complementary, which we denote by $\boldsymbol{x} \sim \boldsymbol{y}$, if

$$\sum_k (x_k x_{k+i} + y_k y_{k+i}) = 2N\delta[i], \ i \in \mathsf{Z}.^1 \tag{1}$$

Note that the definition is symmetric in $\boldsymbol{x}$ and $\boldsymbol{y}$. By a slight abuse of notation we will also call a single sequence $\boldsymbol{x} \in \{\pm 1\}^N$ complementary if there exists another sequence $\boldsymbol{y} \in \{\pm 1\}^N$ such that $\boldsymbol{x} \sim \boldsymbol{y}$.

[1] Throughout the paper we will adhere to the convention that if $\boldsymbol{x}$ is a sequence of length N then $x_k = 0$ if $k \notin \{0, \cdots, N-1\}$. Also, $\delta[i] = 1$ for $i = 0$ and zero otherwise.

N	$\lvert\mathcal{G}_N\rvert$	N	$\lvert\mathcal{G}_N\rvert$
2	4	16	384
4	8	20	272
8	48	26	16
10	32	32	3840

Table 1: The number of Golay sequences for lengths up to 32. The set of Golay sequences of length N is denoted by $\mathcal{G}_N$.

Example 1 The simplest example of a complementary pair is $++ \sim +-$.[2] Examples of complementary pairs of length 4, 8, and 10 are $+++- \sim ++-+$, $+++-+++- \sim +++---+-$, and $-++-+---- \sim +-+---++--$. ◇

An alternative common name for complementary sequences in the literature is Golay sequences. One of the main results of Golay's original work was the following necessary condition on the length of Golay sequences.

Lemma 1 (Golay [2]) A necessary condition for Golay sequences of length N to exist is that N is even and that N can be written as the sum of at most two squares.

Hence, for lengths up to 50 the admissible lengths are $N = 2$, 4, 8, 10, 16, 18, 20, 26, 32, 34, 36, 40 and 50. It was shown by Golay by means of an exhaustive search that there are no Golay sequences of length 18. An algebraic proof of this fact was later given by Kruskal [4] and, more generally, Griffin proved that there are no Golay sequences of length $2 \cdot 9^t$, [5]. Finally, Anders and Stanton [6] gave a computational proof that there are no Golay sequences of lengths 34, 36, and 50. So for lengths up to 50, Golay sequences exist for $N = 2$, 4, 8, 10, 16, 20, 26, 32, 40. Table 1 list the number of Golay sequences of lengths up to 32. Golay also stated various ways of deriving new complementary pairs from a given complementary pair. Given $\boldsymbol{x} \in \mathsf{C}^N$, let $\hat{\boldsymbol{x}}$ denote the "time-reversed" sequence with components $(\hat{\boldsymbol{x}})_k = (\boldsymbol{x})_{N-k-1}$, $0 \leq k < N$. We will also use the shorthand $\hat{\pm}\boldsymbol{x}$ to denote any of the sequences $\boldsymbol{x}$, $-\boldsymbol{x}$, $\hat{\boldsymbol{x}}$, or $-\hat{\boldsymbol{x}}$.

Fact 1 (Golay [2]) Let $\boldsymbol{x} \sim \boldsymbol{y}$. Then

(i) $\hat{\pm}\boldsymbol{x} \sim \boldsymbol{y}$.

(ii) $\boldsymbol{a} \sim \boldsymbol{b}$, where $a_k = (-1)^k x_k$ and $b_k = (-1)^k y_k$, $0 \leq k < N$.

(iii) $\boldsymbol{x}\boldsymbol{y} \sim \boldsymbol{x}(-\boldsymbol{y})$.

(iv) $x_0 y_0 x_1 \cdots x_{n-1} y_{n-1} \sim x_0(-y_0)x_1 \cdots x_{n-1}(-y_{n-1})$.

The first two rules describe ways to derive new complementary pairs of the same length, whereas the remaining rules specify how to derive complementary pairs of twice the

[2]For notational convenience we write $\pm$ instead of ± 1.

original length. Golay also stated a rule of how to derive a complementary pair of length $2NN'$ from two complementary pairs of length N and N', respectively.

To date there is quite an extensive literature on Golay sequences and their generalizations. Welti [7], Turyn [8], and Taki *et al.* [9] investigated the closely related quaternary and even-shift sequences and their relations. Tseng and Liu [10] investigated complementary sets (rather than complementary pairs) and derived some of their basic properties, see also Frank [11]. Sivaswamy [12] extended the idea of complementary sequences to sequences with components on the unit circle.

Besides their original intented use for spectromety, Golay sequences have found several other applications e.g. their use in surface-wave delay lines as suggested by Tseng [13]. These applications are all based on the remarkable spectral properties of Golay sequences. For a given $\boldsymbol{x} \in \{\pm 1\}^N$ let its discrete-time Fourier transform (DTFT) be denoted by $X(e^{j\Omega})$,

$$X(e^{j\Omega}) := \sum_k x_k e^{-j\Omega k}.$$

The Parseval theorem shows that $\max_{\Omega\in[0,2\pi)} |X(e^{j\Omega})| \geq \sqrt{N}$. On the other hand, translating condition (1) into the frequency domain yields

$$|X(e^{j\Omega})|^2 + |Y(e^{j\Omega})|^2 = 2N,\ \Omega \in [0, 2\pi), \tag{2}$$

which implies that all Golay sequences fulfill

$$|X(e^{j\Omega})| \leq \sqrt{2N}. \tag{3}$$

The *crest* factor of $X(e^{j\Omega})$ is defined as its peak-to-average power ratio. From Parsevals theorem and (3) we conclude that if $X(e^{j\Omega})$ is the DTFT of a Golay sequence then

$$1 \leq \frac{\max_{\Omega\in[0,2\pi)} |X(e^{j\Omega})|^2}{\frac{1}{2\pi}\int_0^{2\pi} |X(e^{j\Omega})|^2 d\Omega} = \frac{\max_{\Omega\in[0,2\pi)} |X(e^{j\Omega})|^2}{N} \leq 2,$$

which shows that Golay sequences have a remarkable low crest factor.

Based on their spectral properties, Boyd [14] and Popović [15] suggested the use of Golay sequences for multitone frequency response testing.[3] A potentially even more important application of Golay sequences was suggested by Jones *et. al.* [21]. They suggested their use in multitone systems, [22], to mitigate the inherently large peak-to-average power ratio. The idea is to restrict the input to the individual modulators of the equally spaced carriers to those sequences which result in low crest factors. For simplicity assume that for the time period $t \in [0, T)$ the multitone signal can be modeled as

$$s(t) = \sum_{k=0}^{N-1} x_k e^{2j\pi f_0 kt}$$

where N is the number of modulators, x_k is the input to the kth modulator during this time period, f_0 is the frequency spacing of the modulators, and T is assumed to be a multiple of $1/f_0$. The crest factor of $s(t)$ for $t \in [0, T)$ is a function of the transmitted

[3]Schroeder [16], Popović [15] and Gimlin *et. al.* [17] investigated the use of Newman and related phases to generate low crest factor sequences. Shlien [18], Narahasi *et. al.* [19], and Greenstein *et. al.* [20] suggested the use of optimization programs to find suitable phases.

data $\boldsymbol{x}$. Given a maximum acceptable crest factor it is straightforward to determine by computer the number of all those $\boldsymbol{x} \in \{\pm 1\}^N$ which result in acceptable crest factors. This gives rise to a tradeoff between crest factor and achievable rate (number of admissable sequences). Clearly, the achievable rate is a nondecreasing function in the maximal admissable crest factor. This tradeoff is shown in Figure 1 for a 16 carrier system. From

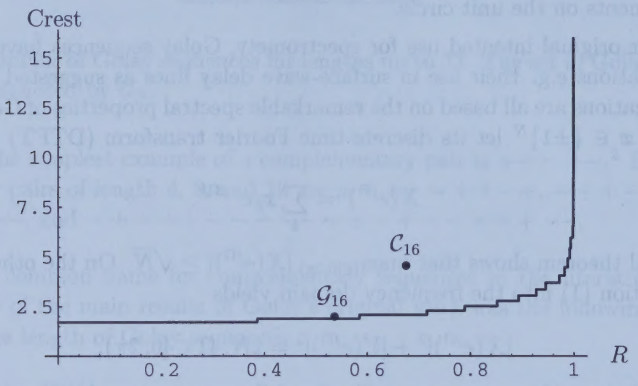


Figure 1: Tradeoff between maximal admissable crest factor and achievable rate for $N = 16$. The rate R is defined as the logarithm to the base 2 of the number of admissable sequences divided by N. The black dots corresponds to the tradeoffs achievable by employing Golay sequences ($\mathcal{G}_{16}$) and cyclic Golay sequences ($\mathcal{C}_{16}$).

our previous discussion we know that if we limit the input to the N modulators to the set of Golay sequences of length N then the resulting maximal crest factor is at most 2. Further, from Table 1 the number of Golay sequences of length 16 is 384 which corresponds to a rate of $R = 0.537$. The corresponding point is also shown in Figure 1. We see from this figure that for a maximal admissable crest factor of 2 and 16 carriers, Golay sequences are almost optimal in terms of the achievable rate.[4] The second point shown in Figure 1 corresponds to the achievable tradeoff by employing cyclic Golay sequences, which are described further below.

There is a large practical advantage in using structured sets of low crest factor sequences such as Golay sequences as opposed to just using the list of low crest factor sequences as found by a computer search. Whereas computer generated lists of sequences have to be encoded, in general, by table-lookup techniques, structured sets of low crest factor sequences admit a more compact description which can be exploited in the encoding as well as decoding stage.

Typically, structured sets of low crest factor sequences have been described as follows. Given the desired set, call it $\mathcal{C}$, let $\mathcal{A}$ be a subgroup of its automorphisms group. Letting $\mathcal{A}$ act on $\mathcal{C}$ partitions $\mathcal{C}$ into equivalence classes. Hence, to describe $\mathcal{C}$ it suffices to

[4]The exact number of sequences of length 16 with a crest factor of at most 2 is 520 which corresponds to a rate of $R = 0.564$.

describe one member out of each equivalence class. These members are often called the *kernels*. Clearly, this procedure is the more efficient the larger $\mathcal{A}$ is.

Example 2 Let $\mathcal{C} = \mathcal{G}_8$, the set of Golay sequences of length 8. We see from (i) and (ii) of Fact 1 that negation, time reversal, and negation of every odd components are automorphisms of $\mathcal{G}_8$. For $\mathcal{A}$ we take the group generated by these maps, where the group operation is concatenation of maps. One can check that in this way $\mathcal{G}_8$ can be described by means of the 6 kernels $\{-++-----, +--+----, -+--+--, --+-+--, ++-++--, +----+--\}$, which generate all 48 elements if we let $\mathcal{A}$ act on them. $\diamond$

In the sequel we will give some alternative ways of deriving compact descriptions of Golay sequences and their extensions. In this paper we limit ourselves to Golay sequences and their cyclic extensions although similar techniques can be used for polyphase sequences and other extensions of Golay sequences. We first show how a fundamental notion of commutative algebra, namely the notion of Gröbner bases, can be successfully employed to give compact description of Golay sequences. We then concentrate on the practically important case of Golay sequences of length 2^n. For those sequences we give a recursive definition of a subset of size $2^n n!$, which we conjecture to coincide with the set of Golay sequences. Finally, we relax our constraint on the sequences to arrive at cyclic Golay sequences. Because of space limitations all proofs are omitted.

2 Main Results

We start by reformulating our problem in terms of a system of multivariable polynomial equations for which a systematic solution approach is known.

Assume that we want to find all Golay pairs $\boldsymbol{x}$ and $\boldsymbol{y}$ of length N. Let

$$p_j(x_0, \cdots, x_{N-1}, y_0, \cdots, y_{N-1}), 1 \leq j \leq 3N-1,$$

be the following set of polynomials.

$$p_j = \begin{cases} x_j^2 - 1, & 0 \leq j \leq N-1 \\ y_{j-N}^2 - 1, & N \leq j \leq 2N-1 \\ \sum_{k=0}^{N-1-j} x_k x_{k+j} + \sum_{k=0}^{N-1-j} y_k y_{k+j}, & 2N \leq j \leq 3N-1. \end{cases} \quad (4)$$

Then $\boldsymbol{x} \sim \boldsymbol{y}$ if and only if $(x_0, \cdots, x_{N-1}, y_0, \cdots, y_{N-1})$ is a solution of the equations $p_j = 0$, $0 \leq j \leq 3N-1$ over C. Fortunately there exists a systematic way of solving polynomial equations. The idea is to transform the given equations into a special form called Gröbner basis, [23, 24, 25, 26], from which the solutions can be read off fairly easily. Algorithms for performing Gröbner bases calculations have been implemented in many popular computer algebra systems like *Mathematica*, [27] or *Macaulay*, [28]. For large equation systems we found the package *Gröbner* by Wolfgang Windsteiger and Bruno Buchberger the most useful, [29].

As example we give a compact description of all Golay sequences of length 16 as derived by this technique.

Lemma 2 $\boldsymbol{x} \in \mathcal{G}_{16}$ if and only if

$$
\begin{aligned}
x_0, x_1, x_2, x_3, x_4, x_5, x_8 &\in \{\pm 1\} \\
x_6 &= \begin{cases} -x_4 x_3 x_1 & x_5 = x_4 x_3 x_2 \\ \{\pm 1\} & x_5 = -x_4 x_3 x_2. \end{cases} \\
x_7 &= x_6 - x_1 x_0 - x_5 x_2 x_0 - x_4 x_3 x_0 \\
x_9 &= \begin{cases} -x_8 x_3 x_2 & x_5 = x_4 x_3 x_2 \\ \{\pm 1\} & x_5 = -x_4 x_3 x_2. \end{cases} \\
x_{10} &= \begin{cases} \frac{1}{2} x_5 (x_9 (x_4 x_3 x_1 - x_6) + x_8 x_3 x_2 (x_4 x_3 x_1 + x_6)) & x_5 = -x_4 x_3 x_2 \\ \{\pm 1\} & x_5 = x_4 x_3 x_2. \end{cases} \\
x_{11} &= -x_{10} x_1 x_0 - x_9 x_2 x_0 - x_8 x_3 x_0 \\
x_{12} &= -x_{10} x_4 x_2 - x_9 x_4 x_1 - x_8 x_6 x_2 - x_8 x_5 x_1 - x_8 x_4 x_3 x_2 x_1 \\
x_{13} &= x_{10} x_4 x_2 x_1 x_0 + x_8 x_6 x_2 x_1 x_0 + x_8 x_4 x_3 x_2 x_0 \\
x_{14} &= x_9 x_4 x_2 x_1 x_0 + x_8 x_5 x_2 x_1 x_0 + x_8 x_4 x_3 x_1 x_0 \\
x_{15} &= -x_8 x_4 x_2 x_1 x_0.
\end{aligned}
$$

Hence, $|\mathcal{G}_{16}| = 384$. Further, each $\boldsymbol{x} \in \mathcal{G}_{16}$ has exactly 4 complements.

The case $\mathcal{G}_{2^n}$ is of particular practical importance and we will now present some analytic results for this case. Given a sequence $\boldsymbol{x} \in \mathsf{C}^N$, where N is even, define the interleaving operator I by means of

$$
(I[\boldsymbol{x}])_k = \begin{cases} x_{k/2} & k \text{ even}, \\ x_{(k+N-1)/2} & k \text{ odd}. \end{cases}
$$

Define the following set of pairs of sequences. Let $\mathcal{P}_1 = \{(+,+)(-,-)\}$. For $n \geq 1$ let $\mathcal{P}_{2^n}$ be defined recursively by

$$
\mathcal{P}_{2^n} := \{(I^k[\boldsymbol{a}\tilde{\boldsymbol{b}}], I^k[\boldsymbol{a}(-\tilde{\boldsymbol{b}})] : (\boldsymbol{a}, \boldsymbol{b}) \in \mathcal{P}_{2^{n-1}}, \tilde{\boldsymbol{b}} \in \{\hat{\pm}\boldsymbol{b}\}\}.
$$

Example 3 $\mathcal{P}_2 = \{(++,+-),(+-,++),(--,-+),(-+,--)\}$. ◇

The next theorem states that these pairs are actually Golay pairs.

Theorem 1 Let $(\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{P}_{2^n}$. Then $\boldsymbol{x} \sim \hat{\pm}\boldsymbol{y}$.

One might be inclined to think that $|\mathcal{P}_{2^n}| = 4n|\mathcal{P}_{2^{n-1}}|$ since the interleaving operator has order n and there are four choices for the argument $\tilde{\boldsymbol{b}}$. Nevertheless, it can be shown that each pair $(\boldsymbol{x}, \boldsymbol{y})$ has exactly two distinct representations $I^k[\boldsymbol{a}\tilde{\boldsymbol{b}}]$ and, hence, $|\mathcal{P}_{2^n}| = 2n|\mathcal{P}_{2^{n-1}}|$. Let $\tilde{\mathcal{G}}_{2^n}$ be defined by

$$
\tilde{\mathcal{G}}_{2^n} := \{\boldsymbol{x} : (\boldsymbol{x}, \boldsymbol{y}) \in \mathcal{P}_{2^n}\}.
$$

We then have

Lemma 3

$$
|\tilde{\mathcal{G}}_{2^n}| = \begin{cases} 4 & n = 1 \\ 2^n n! & n \geq 2. \end{cases}
$$

Conjecture 1 $\mathcal{G}_{2^n} = \tilde{\mathcal{G}}_{2^n}$.

By construction we always have $\mathcal{G}_{2^n} \supseteq \tilde{\mathcal{G}}_{2^n}$. On the other hand, comparing the cardinality of $\tilde{\mathcal{G}}_{2^n}$ as stated in Lemma 3 with the cardinalities of $\mathcal{G}_{2^n}$ as stated in Table 1 we see that Conjecture 1 is true at least up to $n = 5$. As a first small step towards proving Conjecture 1 we prove

Lemma 4 Let $\boldsymbol{x} \in \mathcal{G}_N$, $N = 2^n$ or $N \in 8\mathsf{N}$. Then $\prod_{k=0}^{2^n-1} x_k = 1$.

For some applications a somewhat less stringent condition than (1) might be adequate. We say that $\boldsymbol{x}, \boldsymbol{y} \in \{\pm 1\}^N$ are cyclic Golay sequences if

$$\sum_{k=0}^{N-1} (x_k x_{k+i \bmod N} + y_k y_{k+i \bmod N}) = 2N\delta[i], \ i \in \mathsf{Z}. \tag{5}$$

If $\boldsymbol{X}$ denotes the discrete Fourier transform (DFT) of $\boldsymbol{x}$,

$$X_n = \sum_{k=0}^{N-1} x_k e^{-j\frac{2\pi}{N}kn},$$

then condition (5) can be stated by saying that $\boldsymbol{x} \sim \boldsymbol{y}$ if and only if

$$|X_k|^2 + |Y_k|^2 = 2N, \ k \in \{0, \cdots, N-1\}. \tag{6}$$

We can interpret (6) as a sampled version of (2), where the samples are taken equally spaced in the frequency domain at $\Omega = \frac{2\pi k}{N}$. Let $\mathcal{C}_N$ denote the set of all cyclic Golay sequences of length N. Then we see from the above observation that $\mathcal{G}_N \subseteq \mathcal{C}_N$.

As in the case of acyclic Golay sequences we have

Lemma 5 A necessary condition for cyclic Golay sequences of length N to exist is that N is even and that N can be written as the sum of at most two squares.

Table 2 summarizes $|\mathcal{C}_N|$ for lengths up to 32. The parameter R_N is defined as $R_N := \frac{\log_2 |\mathcal{C}_N|}{N}$. This is the resulting rate (loss) if $\mathcal{C}_N$ is used in a multitone system to reduce the peak-to-average power ratio. Also shown is the crest factor. We note that at least up to length 32, cyclic and acyclic Golay sequences exist for exactly the same lengths.

It can be checked that all statements in Fact 1 are also valid for cyclic Golay sequences. Nevertheless, the class of rules to generate new pairs of cyclic Golay sequences from old pairs of cyclic Golay sequences is strictly larger than for the acyclic Golay sequences. In order to state these rules we introduce the following notation. Given $\boldsymbol{x} \in \mathsf{C}^N$ let S^j, $j \in \mathsf{Z}$, be the cyclic shift operator defined by

$$(S^j[\boldsymbol{x}])_k = x_{k+j \bmod N}.$$

Let σ be a permutation on $\{0, \cdots, N-1\}$ and let $\boldsymbol{P}_\sigma$ be the corresponding permutation matrix, so that the result of σ acting on the components of $\boldsymbol{x}$ is given by $\boldsymbol{x}\boldsymbol{P}$. We say that σ *commutes* with the DFT if $\boldsymbol{F}\boldsymbol{P} = \boldsymbol{P}\boldsymbol{F}$, where $\boldsymbol{F}$ is the DFT matrix. The simplest such permutation is time reversal since given any DFT pair $x_k \leftrightarrow X_n$, $x_{-n} \leftrightarrow X_{-k}$ is also a valid DFT pair. Fortunately, the set of such permutations is well known, [30, 31]. We can now state the additional two rules.

N	$\lvert\mathcal{C}_N\rvert$	R_N	Crest
2	4	1.0	2.0
4	8	0.75	1.77
8	96	0.82312	3.45
10	160	0.732193	3.48
16	1792	0.67546	4.56
20	27520	0.73741	5.48
26	111480	0.644863	5.99
32	1154048	0.629321	6.70068

Table 2: The cardinality $|\mathcal{C}_N|$ for lengths up to 32 as well as the rate $R_N := \frac{\log_2 |\mathcal{C}_N|}{N}$ and the crest factor.

Lemma 6 Let $\boldsymbol{x} \sim \boldsymbol{y}$, $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{C}_N$. Then

(i) $S^{j_1}[\boldsymbol{x}] \sim S^{j_2}[\boldsymbol{y}]$, $j_1, j_2 \in \mathbb{Z}$.

(ii) $\boldsymbol{x}\boldsymbol{P}_\sigma \sim \boldsymbol{y}\boldsymbol{P}_\sigma$ for any permutation σ that commutes with the DFT.

Example 4 For $N = 8$ one can check that there are four permutations that commute with the DFT. These are $\sigma_1 = \{0,1,2,3,4,5,6,7\}$, $\sigma_2 = \{0,7,6,5,4,3,2,1\}$, $\sigma_3 = \{0,3,6,1,4,7,2,5\}$, and $\sigma_4 = \{0,5,2,7,4,1,6,3\}$, where a j at the ith position means that i gets mapped into j. Note that σ_1 and σ_2 as well as σ_3 and σ_4 are time reversals of each other (up to a cyclic shift). ◇

One can use the rules of Fact 1 as well as Lemma 6 to gain a compact description of $\mathcal{C}_N$ as described in the introduction.

Example 5 One can check that $\mathcal{C}_8$ is the union of the four sets generated by applying our rules to the kernels $++++++--$, $+++++-+-$, and $+++---+--$. The sizes of the corresponding equivalence classes are 64, 16, and 16, respectively. ◇

As in the case of the acyclic Golay sequences, we can again use Gröbner bases techniques to get a compact description of cyclic Golay sequences of moderate length. For $N = 2, 4$ we have $\mathcal{C}_2 = \mathcal{G}_2$ and $\mathcal{C}_4 = \mathcal{G}_4$. As a nontrivial example we deal with the case $\mathcal{C}_8$ in the following lemma.

Lemma 7 $\boldsymbol{x} \in \mathcal{C}_8$ if and only if

$$\begin{aligned} x_0, x_1, x_2, x_3, x_4, x_5 &\in S_2 \\ x_6 &= \begin{cases} -x_4x_2x_0 & x_5 = x_4x_1x_0 \\ S_2 & x_5 = -x_4x_1x_0. \end{cases} \\ x_7 &= -x_6x_3x_2 - x_5x_3x_1 - x_4x_3x_0. \end{aligned}$$

Hence, $|\mathcal{C}_8| = 96$.

3 Acknowledgement

We would like to thank Andrew Odlyzko, Tom Richardson, and Emre Telatar for useful discussions and Emre Telatar for lending us his UNIX expertise. We would also like to thank Wolfgang Windsteiger for his help in installing his package GROEBNER.

References

[1] M. J. E. Golay, "Static multislit spectrometry and its application to the panoramic display of infrared spectra," *J. Opt. Soc. Amer.*, vol. 41, pp. 468–472, July 1951.

[2] M. J. E. Golay, "Complementary sequences," *IEEE Trans. Inform. Theory*, vol. IT–7, pp. 82–87, Apr. 1961.

[3] M. J. E. Golay, "Multislit spectrometry," *J. Opt. Soc. Amer.*, vol. 39, pp. 437–444, June 1949.

[4] J. B. Kruskal, "Golay's complementary series," *IEEE Trans. Inform. Theory*, vol. IT–7, pp. 273–276, Oct. 1961.

[5] M. Griffin, "There are no Golay complementary sequences of length $2 \cdot 9^t$.," *Aequationes Math.*, vol. 15, pp. 73–77, 1977.

[6] T. H. Andres and R. G. Stanton, "Golay sequences," in *Combinatorial Mathematics*, Lecture Notes in Mathematics, vol. 622, Springer-Verlag, 1977.

[7] G. Welti, "Quaternary codes for pulsed radar," *IEEE Trans. Inform. Theory*, vol. IT–6, pp. 400–408, June 1960.

[8] R. Turyn, "Ambiguity functions of complementary sequences," *IEEE Trans. Inform. Theory*, vol. IT–9, pp. 46–47, Jan. 1963.

[9] Y. Taki, H. Miyakawa, M. Hatori, and S. Namba, "Even-shift orthogoanl sequences," *IEEE Trans. Inform. Theory*, vol. IT–15, pp. 295–300, Mar. 1969.

[10] C.-C. Tseng and C. L. Liu, "Complementary sets of sequences," *IEEE Trans. Inform. Theory*, vol. IT–18, pp. 644–652, Sept. 1972.

[11] R. L. Frank, "Polyphase complementary codes," *IEEE Trans. Inform. Theory*, vol. IT–26, pp. 641–647, Nov. 1980.

[12] R. Sivaswamy, "Multiphase complementary codes," *IEEE Trans. Inform. Theory*, vol. IT–24, pp. 546–552, Sept. 1978.

[13] C.-C. Tseng, "Signal multiplexing in surface-wave delay lines using orthogonal pairs of Golay's complementary sequences," *IEEE Trans. Inform. Theory*, vol. IT–17, pp. 103–107, Apr. 1971.

[14] S. Boyd, "Multitone signals with low crest factor," *IEEE Trans. Circuits Syst.*, vol. CAS–33, pp. 1018–1022, Oct. 1986.

[15] B. M. Popović, "Synthesis of power efficient multitone signals with flat amplitude spectrum," *IEEE Trans. Commun.*, vol. COM–39, pp. 1031–1033, July 1991.

[16] M. R. Schroeder, "Synthesis of low-peak-factor signals and binary sequences with low autocorrelation," *IEEE Trans. Inform. Theory*, vol. IT–16, pp. 85–89, Jan. 1970.

[17] D. R. Gimlin, "On minimizing the peak-to-average power ratio for the sum of n sinusoids," *IEEE Trans. Commun.*, vol. COM–41, pp. 631–635, Apr. 1993.

[18] S. Shlien, "Minimization of the peak amplitude of a waveform," *Electronics Letters*, vol. 30, pp. 2098–2099, Dec. 1994.

[19] S. Narahashi and T. Nojima, "New phasing scheme of n-multiple carriers for reducing peak-to-average power ratio," *Electronics Letters*, vol. 30, pp. 1383–1384, Aug. 1994.

[20] L. J. Greenstein and P. J. Fitzgerald, "Phasing multitone signals to minimize peak factors," *IEEE Trans. Commun.*, vol. COM–29, pp. 1072–1074, July 1981.

[21] A. E. Jones, T. A. Wilkinson, and S. K. Barton, "Block coding scheme for reduction of peak to mean envelope power ratio of muticarrier transmission schemes," *Electronics Letters*, vol. 30, pp. 2098–2099, Dec. 1994.

[22] J. A. C. Bingham, "Multicarrier modulation for data trnasmission: An idea whose time has come," *IEEE Comm. Mag.*, pp. 5–14, May 1990.

[23] B. Buchberger, "Gröbner bases - an algorithmic method in polynomial ideal theory," in *Multidimensional Systems Theory* (N. Bose, ed.), ch. 6, J. D. Reidel, 1985.

[24] W. W. Adams and P. Loustaunau, *An Introduction to Gröbner Bases*, vol. III. American Mathematical Society, Graduate Studies in Math., 1994.

[25] J. L. D. Cox and D. O'Shea, *Ideals, Varieties, and Algorithms.* New York: Springer Verlag, 1991.

[26] T. Becker and V. Weispfenning, *Gröbner Bases: A Computational Approach to Commutative Algebra.* New York: Springer Verlag, 1991.

[27] S. Wolfram, *Mathematica: A System for Doing Mathematics by Computer.* New York: Addision-Wesely, Inc., 1991.

[28] D. Bayer and M. Stillman, "Macaulay: A computer algebra system for algebraic geometry." available by anonymous ftp from `zariski.harvard.edu`.

[29] W. Windsteiger and B. Buchberger, "Gröbner: A library for computing Gröbner bases based on SACLIB." available by anonymous ftp from `melmax.risc.uni-linz.ac.at`, 1993.

[30] R. Urbanke, Q. Li, and B. Rimoldi, "On permutations that commute with the discrete Fourier transform," in *Proceedings of the 32nd Allerton Conference on Communication, Control, and Computing*, (Monticello, IL), pp. 230–234, Sept. 28-30 1994. Invited paper.

[31] P. J. S. G. Ferreira, "A group of permutations that commute with the discrete fourier transform," *IEEE Transactions on Signal Processing*, pp. 444–445, Feb. 1994.

Classes of Costas and Sonar Sequences for Multiple-target Recognition

Oscar Moreno[1]
Department of Mathematics and Computer Science,
University of Puerto Rico,
Rio Piedras, Puerto Rico, 00931.
and
Svet Marić
Department of Electrical Engineering
University of Cambridge
Cambridge CB2 1PZ, U.K.

Abstract: The problem we address is that of obtaining the velocity and the distance from us of one or several targets. The case of only one target has been developed by S. W. Golomb and H. Taylor in 1982 and 1984 and also by O. Moreno, R. A. Games and H. Taylor in 1993. In this paper, we develop new classes of Costas and Sonar Sequences for multiple-target recognition, different from those previously given by Moreno and Marić in 1995.

1 Introduction

Costas Sonar sequences were introduced in [1] and in [10] to deal with the following problem: "An object is moving towards (or away) from an observer, and he wants to know effectively the distance from the object and its velocity." The solution to the problem is obtained using the Doppler effect, that states the following: when a signal hits a moving target its frequency changes in direct proportion to the velocity. In other words, if the observer sends out a signal towards a moving target the change between the original frequency and the returned frequency will allow him to determine the velocity of the target, and the time it took to make the round trip will allow him to determine the distance.

However, since the world is noisy, a signal that was sent out might not return. Consequently, you send out l signals with frequencies ranging from 1 to n, at times ranging from 1 to l. If $n = l$ we call it a Costas sequence. Once the whole pattern of signals has returned, the velocity and the distance of the object can be determined as mentioned above. On the other hand, if some of the signals do not return, some ambiguities will arise in the interpretation of what the overall pattern was.

Costas and Sonar sequences are those patterns for which you send out exactly one signal at every time slot, and with the special property that even if only two signals return you can reconstruct the whole pattern. This last point means that there is no ambiguity. The goal involved in constructing sonars is, given n to obtain an n by l sequence for l as large as possible.

Our constructions in this paper produce families of classes of sequences that fall in what are called extended sonar and Costas sequence (see [9]). In the case of ordinary sonar and Costas sequence exactly one signal at every time is sent, on the other hand

[1]Work partially supported by NSF grant CDA-9417362

in the case of extended sonar sequences *at most* one signal is sent at every time, or in other words, one can choose not to send any signal in some time slots. The problem for this application is given n frequencies, construct and n by m sequence for m as large as possible. As shown in [9] with the extended sonar sequence one can obtain larger sequences with the same range of frequencies than with the ordinary sonar sequence. This last point also means that one can use the same equipment for both sequences making the extended sonar sequence cost effective.

The paper is organized as follows. In the next section, for completeness, we first give some necessary definitions and also the constructions of Welch (WC) and Lempel-Golomb (LC) arrays, and quadratic and hyperbolic congruence codes. In the third section we give the new constructions.

2 Preliminaries

Costas and Sonar sequences in general are divided in two groups based on the implemented signal processing at the receiver. In his original work J. Costas [2] pointed out the importance of the sequences (here called Costas sequences) having one and only one pulse per each time slot and frequency channel. We also call this type of sequence a full sequence.

Definition 1: [3] A Costas array is an $N \times N$ permutation matrix with the property that vectors connecting two 1's are all distinct vectors. (That is no two vectors are equal in both amplitude and slope.)

An extensive study of the algebraic constructions of Costas arrays is given in [6] and [3]. All known constructions producing a large number of Costas arrays of equal length are based on the properties of finite fields and primitive roots.

Theorem 1 (Welch):[2] Let α be a primitive root of an odd prime p. Then the array with

$$a_{k,y(k)} = 1,\ 1 \leq k,\ y(k) \leq p-1 \tag{1}$$

if and only if

$$y(k) = \alpha^k\ (mod\, p),\ 1 \leq k \leq p-1, \tag{2}$$

otherwise $a_{k,y(k)} = 0$ is a Costas array.

Note that the extensions of this construction have been found by Golomb and Taylor [4], see also Moreno [7].

Theorem 2 (Lempel-Golomb): Let α and β be primitive roots of the finite field, $GF(q)$, $q = p^n$, p a prime. Then the $(q-2) \times (q-2)$ permutation matrix with $a_{k,y(k)} = 1$ if and only if

$$y(k) = log_\beta(1 - \alpha^k),\ 1 \leq k \leq (q-2) \tag{3}$$

is a Costas array.

It is important to note the difference between the finite fields in the two theorems. Theorem 1 is valid only over $GF(p)$ whereas theorem 2 is valid also for $GF(p^n)$ which then requires the representation of the elements through polynomials.

By definition the number of hits in the 2-D auto-correlation function of Costas arrays is ideal. However the multi-user environment and rather poor 2-D cross-correlation properties of Costas arrays necessitated the construction of other sequence families like the n-th order congruence codes.

[2]The proofs of the theorems in this sections can be found in [6] or [3].

Definition 2: Let $GF(p)$ denote the finite field over a prime p, and let $\tilde{GF}(p) = GF(p) - \{0\}$. Then the n-th order congruence sequence of length p is defined by:

$$y(k) = \alpha k^n \, (mod\, p) \, 0 \leq k \leq p \; and \; some \; \alpha \in \tilde{GF}(p) \tag{4}$$

Theorem 3: An n-th order congruence sequence constructed over the field of prime p, is a full sequence when $d = (\phi(p), n) = 1$ where ϕ denotes the Euler ϕ function and $d = (x, y)$ is the greatest common divisor of x and y.

Theorem 4: A n-th order congruence sequence $(n > 1)$ of length p has at most $n-1$ hits in its 2-D auto-correlation function except for the (0,0) shift, and at most n hits in its 2-D cross-correlation function formed with any other sequence from the same family.

Definition 3: Hyperbolic sequence for finite field $GF(p)$ is defined as:

$$y(k) = \alpha/k \, (mod\, p) \, k = 1, 2, \ldots, p-1 \; and \; some \; \alpha \neq 0 \in GF(p) \tag{5}$$

Theorem 5: Hyperbolic sequences of length $p-1$ have at most two hits in its auto-correlation function except for the (0,0) shift, and at most two hits in its 2-D cross-correlation function formed with any other code from the same family.

Consider as before α, β primitive roots of the finite field GF(q) and $P_n(x)$ the set of all polynomials of degree $\leq n$ over GF(q). If $f(x) \in P_n(x)$ we will define an equivalence relation so that $f(x)$ is equivalent to $bf(ax)$ where b, a are arbitrary non-zero elements of GF(q). Consider now $E_n(x)$ a subset of $P_n(x)$ containing a set of representatives of the equivalence relation, excluding all the polynomials which are non-trivially self-equivalent. In other words any element of $E_n(x)$ is not equivalent to any other element of $E_n(x)$, and is also not self-equivalent. Furthermore $E_n(x)$ is as large as possible given the above mentioned properties under the equivalence relation.

We define now

$$y_f(k) = \log_\beta f(\alpha^k) \qquad 1 \leq k \leq q-1$$

where $f \in E_d(x)$ with α, β primitive elements over GF(q). Note that the array defined using y_f must have empty columns whenever $f(\alpha^i) = 0$, and therefore falls in the category of extended patterns define in [9]. With the above comments we have the main result of [9]:

Theorem 6: The codes defined by y_f for $f \in E_d(x)$ have at most d hits in its 2-D auto-correlation and 2-D cross-correlation function.

Remark: Note that $E_d(x)$ has approximately q^{d-1} elements.

3 New Constructions

Construction I: Generalization of the hyperbolic sequences from a prime field to an arbitrary finite field. Let α be a primitive element in the finite field $GF(q)$, $q = p^m$, define

$$y(k) = \log_\alpha\left(\frac{\alpha^k - A}{\alpha^k - 1}\right) \quad k = 1, 2, \ldots, q-2.$$

and $A \neq 0, 1$ in $GF(q)$. Note that y is an extended sequence, since when $\alpha^k = A$ we define $y(k)$ as blank. We will call those sequences generalized hyperbolic sequences.

Theorem 7: Generalized hyperbolic sequences of length $q-2$ (including a blank) have at most two hits in its auto-correlation function, and at most two hits in its 2-D cross-correlation function formed with any other sequence from the same family.

Construction II: Extended hyperbolic sonar sequences over a prime field. We define an extended hyperbolic sonar sequence as:

$$y(k) \equiv \frac{a}{k} + ck \pmod{p} \quad k = i, i+1, i+2, \ldots, i+p+1 \quad a, c \in GF(q).$$

If we define all the $y(k)$ with the same a to be equivalent, and if we pick exactly one representative from each equivalence class in our next theorem we have:

Theorem 8: Extended Hyperbolic sonar sequences of length $p+1$ (including a blank) have at most two hits in its auto-correlation function, and at most two hits in its 2-D cross-correlation function formed with any other representantive sequence with a different value of a.

References

[1] S. W. Golomb and H. Taylor, *Two -Dimensional Synchronization Patterns for Minimum Ambiguity*, IEEE Transactions on Information Theory, 28:4 (1982), pp. 600–604.

[2] J. P. Costas. A Study of a Class of Detection Waveforms Having Nearly Ideal Range-Doppler Ambiguity Properties. In *Proceedings of the IEEE, vol 72, no 8* pp 996-1009, August 1984.

[3] S.W. Golomb and H. Taylor. Construction and Properties of Costas arrays. In *Proceedings of the IEEE, vol. 72, no. 9* September 1984.

[4] S.W. Golomb and H. Taylor. Two dimensional synchronization patterns for minimum ambiguity. In *"IEEE Trans. Information Theory, vol. IT-28 no. 4* July, 1982.

[5] S. V. Maric and I. Seskar and E. L. Titlebaum. On cross-ambiguity Properties of Welch-Costas arrays. In *"IEEE Transactions on Aerospace and Electronic Systems*, October 1994.

[6] S.V. Maric and E.L. Titlebaum. A Class of Frequency Hop Codes with Nearly Ideal Characteristics for Use in Multiple Access Spread Spectrum Communications and Radar and Sonar Systems In *IEEE Transactions on Communications,* September 1992.

[7] O. Moreno. A shifting property of some Costas sequences. In *Ars Combinatoria* June, 1994.

[8] O. Moreno, R. A. Games and H. Taylor, *Sonar Sequences from Costas arrays and the best known sonar sequences with up to 100 symbols*, IEEE Transactions on Information Theory, 39:6 (1993), pp. 1985–1989.

[9] O. Moreno, S. W. Golomb and C. J. Corrada. Extended Sonar sequences. Pre-print.

[10] O. Moreno, S.V. Marić. A Class of Frequency Hop Codes with Nearly Ideal Characteristics for Multiple-target Recognition. In *Thirty-third Annual Allerton Conference on Communication, Control, and Computing,* October 1995, pp. 911–914.

Dense Mordell-Weil Lattices of Elliptic Curves Over Function Fields: Explicit Computation[1]

Vahid Tarokh[2] and Ian F. Blake [3]

Abstract

It was recently observed the Mordell-Weil lattices of certain elliptic curves give record packing densities in many dimensions. It is useful to have explicit bases for these lattices. In this note, Gram matrices for some of these lattices are computed.

1 Introduction

A stimulating problem in mathematics is to compute the maximum possible densities of sphere packings in the Euclidean space $\mathbb{R}^n$. This is intimately related to the design of efficient communication systems. In this note, we briefly discuss the computation of dense lattices constructed by Shioda and later by Elkies.

Let K denote the function field $k(t)$ of a finite field k of all the rational functions in a variable t. let E be an elliptic curve over K. Then E has a realization as a surface whose fibres are elliptic curves over k. The Mordell-Weil Theorem [5] indicates that the group $E(K)$ of K-rational points of E, modulo its torsion subgroup $E_{tor}(K)$, is a finitely generated free abelian group. By the Neron-Tate Theorem [5] the canonical height induces a positive definite bilinear symmetric form on $(E(K)/E_{tor}(K)) \otimes_{\mathbb{Z}} \mathbb{R}$. This gives $(E(K)/E_{tor}(K)) \otimes_{\mathbb{Z}} \mathbb{R}$ the structure of a lattice L, referred to as the *Mordell-Weil lattice* of E [5].

To this date, in studying elliptic curves, the main focus has been analyzing the structure of Mordell-Weil groups of elliptic curves defined over the rational field $\mathbb{Q}$. These studies were originated in some sense by Fermat and have motivated a huge body of literature that has lead to invention of many new branches of mathematics.

Recently, Shioda [4] studied a family of elliptic curves defined over function fields of characteristic $\equiv 5 \bmod 6$. It turned out that the Mordell-Weil lattices of the curves in this family produce lattice packings that improve on the previous density records in some dimensions. This line of research was continued by Elkies who discovered another family of curves with similar properties [2]. In this short note, we provide generator matrices for some of these lattices. To this end, we briefly study the Elkies and Shioda families of elliptic curves. In the following discussion, the length of a vector $\mathbf{v} \in \mathbb{R}^n$ is denoted by $\|\mathbf{v}\|$. The *norm* of $\mathbf{v}$ is then equal to $\|\mathbf{v}\|^2$.

Definition 1 *Fix a positive integer n, let $q = 2^n$ and k denote the finite field $GF(q^2)$. Let*

$$E: \ y^2 + y = x^3 + t^{q+1} + \beta, \tag{1}$$

[1]This work was supported in part by the National Sciences and Engineering Research Council of Canada grant number A7382.

[2]V. Tarokh was with the Department of Electrical and Computer Engineering, University of Waterloo, Ontario N2L 3G1, Canada. He is now with AT&T Labs, Room 2D-155, 600 Mountain Avenue, Murray Hill, N.J. 07974.

[3]Ian Blake was with the Department of Electrical and Computer Engineering, University of Waterloo, Waterloo, Ontario N2L 3G1, Canada. He is now with Hewlett Packard Laboratories. 1501 Page Mill Road, Palo Alto, CA 94304.

defined over $K = k(t)$, where $\beta = 0$ for n odd. For n even, let β be an arbitrary element of k whose trace into $GF(2)$ is 1. For $n \in \mathbb{N}$, the above equation defines the Elkies family of elliptic curves.

Definition 2 *Let*

$$E_0 : y^2 = x^3 + 1 + t^{p+1} \tag{2}$$

defined over the ground field $GF(p^2)(t)$, where p is a prime congruent to 5 modulo 6. For different values of p, the above equation defines the Shioda family of Elliptic curves.

Some of the results of Elkies and Shioda are summarized in the following Theorems [2, 4].

Theorem 1 *Let E be an Elkies elliptic curve defined as above. Let L be the Mordell-Weil lattice of E.*

- *The elements $(x, y) \in E(K)$ with $x = a[t]/b[t]$ with $a[t]$ and $b[t]$ relatively prime polynomials satisfying $\deg(a[t]) - \deg(b[t]) > (q+1)/3$ form a sublattice L_0 of L. Furthermore, L_0 is an even lattice of dimension $2q$.*
- *For n even, L_0 is equal to L and for odd n, L_0 has index 4 in L.*
- *The minimum norm of L_0 is not less than $(q+1)/3$. For n even, the determinant of L_0 is not greater than $q^{2[q/6]}$, and for n odd this determinant is not greater than $4q^{2[q/6]}$, where $[x]$ denotes the closest integer to x.*
- *L_0 has center density*

$$\delta(L_0) \geq q^{-(q-4)/6}\left(\frac{q+2}{12}\right)^q$$

for n even, and

$$\delta(L_0) \geq \frac{1}{2} q^{-(q-2)/6}\left(\frac{q+4}{12}\right)^q$$

for n odd.

Theorem 2 *Let E_0 be a Shioda elliptic curve as given by (2).*

- *The Mordell-Weil lattice of $E(K)$ is then an even integral lattice L of dimension $2p-2$, minimum norm $(p+1)/3$, with $\det(L) = p^{(2x-2)}$, where $x = (p+1)/6$.*
- *The center density of L is*

$$\delta(L) = \frac{\left(\frac{p+1}{12}\right)^{p-1}}{p^{(p-5)/6}}.$$

- *The kissing number τ of the lattice L satisfies the bound*

$$\tau \geq 6p(p-1).$$

From the above results, it is immediately observed that Elkies lattices improve on the best packing bounds in dimensions 64, 128, 256, 512, 1024 , 2048 and 4096[2]. Shioda's lattices also give record density in dimensions 80, 104, 508 and 520 [4]. Elkies also observed that the lattice E_{32} associated with (1) in dimension 32 achieves the same record density as Q_{32}. In fact, it has the same theta series as Q_{32}. Interestingly, E_{32}^*, the dual of this 32 dimensional lattice, has also the same center density but is not isomorphic to E_{32}. Thus E_{32} is not iso-dual [2].

In general, it is not easy to find generator matrices for Mordell-Weil lattices. In fact, in it's full generality, this is a long standing open problem. In light of this unfortunate fact, we restrict attention to Elkies and Shioda lattices.

2 The Neron-Tate Height

Let E be an elliptic curve over the function field $K = GF(p^n)(t)$ for some prime p. If $P = (x, y) \in E(K)$ with $x = a[t]/b[t]$, where $a[t]$ and $b[t]$ are relatively prime polynomials, then $H(P) = \max(\deg(a[t]), \deg(b[t]))$ is the *naive height* of P. When P is the point at infinity then $H(P)$ is defined to be zero. The canonical height of P is defined by [5]

$$N(P) = \lim_{m \to \infty} \frac{H(2^m P)}{2^{2m}},$$

where $2^m P$ is computed in the Mordell-Weil group of E.

The Neron-Tate theorem states that

$$P \cdot Q = \frac{N(P+Q) - N(P) - N(Q)}{2}$$

induces a positive definite symmetric bilinear form on $(E(K)/E_{tor}(K)) \otimes_{\mathbb{Z}} \mathbb{R}$.

In computing the Mordell-Weil lattices of Shioda and Elkies curves, it is useful to compute the canonical height. The following theorem of [6] gives the canonical height for a Shioda curve.

Theorem 3 *Let E be a Shioda elliptic curve and $P = (x, y) \in E(K)$ with $x = a[t]/b[t]$ where $a[t]$ and $b[t]$ are relatively prime polynomials. The canonical height of P is given by*

$$N(P) = \begin{cases} \deg(a[t]) & \text{if } \deg(a[t]) \geq \deg(b[t]) + \frac{(p+1)}{3}, \\ \deg(b[t]) + \frac{(p+1)}{3} & \text{otherwise.} \end{cases} \tag{3}$$

The following theorem of Elkies [2] establishes a similar result for an Elkies curve.

Theorem 4 *For an Elkies elliptic curve, the canonical height and the naive height coincide.*

Corollary 1 *The Mordell-Weil groups of Elkies and Shioda curves are torsion free.*

Proof: We refer the reader to [6]. □

Corollary 2 *The minimum norm of the Mordell-Weil lattice of a Shioda curve is $\frac{(p+1)}{3}$.*

Proof: If P is a point of the Mordell-Weil group of a Shioda curve other than the point at infinity, then by (3) $H(P) \geq \frac{(p+1)}{3}$. Thus to establish the result, it suffices to give one point of the lattice having such norm. By Theorem 3 the point $(-1, t^{(p+1)/2})$ is of this norm. □

3 Explicit Computation

Having the Neron-Tate height computed, it is possible to compute the Mordell-Weil lattices of Shioda and Elkies curves. The idea is to look for a basis of the lattice whose longest vector attains a reasonable minimum norm. For the lengths of the vectors of a basis of a lattice, one usually appeals to the following theorem of Hermite.

Theorem 5 *Let L be a lattice of dimension n and volume $V(L)$. Then there exists a basis $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ of L satisfying*

$$\|\mathbf{v}_1\| \, \|\mathbf{v}_2\| \ldots \|\mathbf{v}_n\| \leq (2/\sqrt{3})^{n(n-1)/2} V(L). \tag{4}$$

Proof: The reader is referred to [3]. □

It turns out that appealing to Theorem 5 causes excruciating computational difficulties in determining a Gram matrix for L. On the other hand, it is known that the densest lattices in $\mathbb{R}^n$ have *equivalent successive minima.* Thus, it seems natural to look for an independent set of vectors $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ in the first couple of shells of L. We have used this idea and determined a set of independent vectors $\{\mathbf{v}_1, \mathbf{v}_2, \ldots, \mathbf{v}_n\}$ of L. Using a random algorithm [6], this set of independent vectors was then extended to contain a basis of L. This method was then employed to explicitly compute bases of some of the members of Elkies and Shioda families. For the details of this computation, we refer the reader to [6]. As an example, we provide a Gram-Schmidt of the 32-dimensional Elkies lattice here (see next page). Another example is the 64-dimensional Elkies lattice E_{64}. A Gram matrix for E_{64} can be obtained using the World-Wide-Web address:

http://www.research.att.com/~njas/lattices.

Acknowledgement: We sincerely thank N. Sloane and G. Nebe for setting up the above World-Wide-Web address that also contains information about many other interesting lattices.

References

[1] J.H. Conway and N.J.A. Sloane, *Sphere Packings, Lattices and Groups*, New York: Springer-Verlag, 1993.

[2] N.D. Elkies, Mordell-Weil lattices in characteristic 2, I: Construction and basic properties, preprint.

[3] S. Lang, Conjectured Diophantine estimates on elliptic curves, Papers dedicated to I.R. Shafarevich on the occasion of his sixtieth birthday, Vol I, Birkhauser Boston Inc., 1983.

[4] T. Shioda, Mordell-Weil lattices and sphere packing, *American Journal of Mathematics* 113 , pp. 931-948, 1991.

[5] J. H. Silverman, *The Arithmetic of Elliptic Curves*, New York: Springer 1986.

[6] V. Tarokh, *Trellis Complexity Versus the Coding Gain of Lattice-Based Communication Systems*, Ph.D. thesis, The University of Waterloo, Ontario, Canada 1995.

A Gram Matrix of E_{32}.

Buffer Overflow Probabilities in Multiclass ATM Switches *

Dimitris Bertsimas[†] dbertsim@aris.mit.edu
Ioannis Ch. Paschalidis[‡] yannisp@bu.edu
John N. Tsitsiklis[§] jnt@mit.edu

June 1996

Abstract

In this paper we consider a multiclass multiplexer, with segregated buffers for each type of traffic. Under two specific scheduling policies for sharing bandwidth: the *generalized processor sharing policy (GPS)* and the *generalized longest queue first policy (GLQF)*, we seek the asymptotic tail of the buffer overflow probability for each buffer. We assume dependent arrival and service processes to model bursty traffic. In the standard *large deviations* methodology, we provide a lower and a matching (up to first degree in the exponent) upper bound on the buffer overflow probabilities. We relate the lower bound derivation to a *deterministic optimal control problem*, which we explicitly solve. Optimal state trajectories of the control problem correspond to typical congestion scenarios. We explicitly and in detail characterize the *most likely* modes of overflow. We find that the GLQF policy outperforms the GPS policy with respect to loss probabilities characteristics. The performance of strict priority policies is obtained as a corollary. Our results have important implications in traffic management of high-speed networks and can be used as a basis for an admission control mechanism which guarantees different loss probability for each type of traffic.

Keywords: Communication networks, ATM-based B-ISDN, Large Deviations.

1 Introduction

Future high speed, packet-switched communication networks, for example ATM-based B-ISDN networks, will accommodate various types of traffic, namely, digitized voice, encoded video, and data. One of the central and most challenging current problems in computer networking is the design and the operation of these networks.

Congestion causes packet losses, due to buffer overflows, that greatly contribute to the degradation of the *quality of service (qos)*. We quantify qos by the buffer overflow probability. It is desirable to operate the network in a regime where packet loss probabilities are very small, e.g., in the order of 10^{-9}. An essential step for preventing congestion through a variety of control mechanisms (buffer dimensioning, admission control, resource allocation) is to determine how it occurs and to estimate the probabilities of congestion phenomena. The problem is particularly hard since it essentially requires finding the distributions of waiting times and queue lengths in a multiclass network of G/G/1 queues with correlated arrivals (bursty traffic) and non-exponential services. In

*A preliminary version of these results was reported in [2]. The results in this paper are included in [14]. The full paper appears in [3, 4].

Research supported by a Presidential Young Investigator award DDM-9158118 with matching funds from Draper Laboratory, and by the ARO under grant DAAL-03-92-G-0115.

[†]Sloan School of Management, MIT, Cambridge, MA 02139.

[‡]Department of Manufacturing Engineering, Boston University, 15 St. Mary's St., Boston MA 02215.

[§]Laboratory for Information and Decision Systems, MIT, Cambridge, MA 02139.

this light, it is natural to focus on the *large deviations regime* and obtain asymptotics of the congestion probabilities.

In this paper we focus at a simplified version of the problem which retains the most salient features, that is, it is multiclass and has correlated arrival and service processes. In particular, we consider a multiclass multiplexer, with segregated buffers. We seek the asymptotic (as the buffer size goes to infinity) tail of the buffer overflow probability for each buffer. We consider two scheduling policies for sharing bandwidth: the *generalized processor sharing policy (GPS)* ([13]), and the *generalized longest queue first policy (GLQF)*. The GLQF policy is a generalization of the *longest queue first policy (LQF)*. Both are parametric policies and for specific values of the parameters reduce to strict priority policies. Thus, the performance of strict priority policies is obtained as a corollary (approximate results for priority policies appeared in [10]). We compare the loss probabilities characteristics of the GLQF and the GPS policy and find that the first outperforms the second.

In the standard *large deviations* methodology we provide a lower and a matching (up to first degree in the exponent) upper bound on the buffer overflow probabilities. We prove that overflows occur in one of two *most likely* ways (modes of overflow) and we explicitly and in detail characterize these modes. We address the case of multiplexing two different traffic streams; for the general case of N streams our lower bound approach (which also determines the modes of overflow) can be easily extended. Our results have important implications in traffic management of high-speed networks. They can be used as a basis for an admission control mechanism which guarantees the desirable loss probability, and allows for different qos requirements for each type of traffic (see [14]).

Large deviations techniques have been applied recently to a variety of problems in communications. The problem of estimating buffer overflow probabilities in a single class queue has received extensive attention (see [14] for literature review). The extension of these ideas to single class networks, although much harder, has been treated in various versions and degrees of rigor (see [1] for a literature review of network results) . Results for the GPS policy with deterministic service capacity have appeared in [7, 16, 12, 6]. In [11] the use of the LQF policy in high speed networks is suggested, and buffer sizes that guarantee no loss are calculated using a deterministic model.

In this paper, we provide an *optimal control formulation* of the problem both in the GPS and the GLQF. In particular, the exponent of the overflow probability is the optimal value of the control problem, which we explicitly solve. Optimal state-trajectories of the control problem correspond to the most likely modes of overflow; from the solution of the control problem we obtain a detailed characterization of these modes. This formulation is general enough to include any scheduling policy. The only thing that changes with the policy is the dynamics of the system. Moreover, our work extends the GPS results in the literature to the case of stochastic service capacity, and provides results for the GLQF policy. The stochasticity in the capacity makes it possible to treat more complicated service disciplines (mixed priority and GPS or GLQF). Notice that this stochasticity significantly alters the way overflows occur [1].

Regarding the structure of this paper, we begin in Section 2 with some preliminaries and we state a set of modeling assumptions. In Section 3 we define the multiclass model, along with the GPS and the GLQF policy, and we provide an outline of the methodology that we follow. In Section 4 we introduce the optimal control formulation and solve the control problem, thus, establishing both a lower bound on the overflow probabilities and characterizing the most likely modes of overflow. We gather our main results in Section 5, where we also treat the special case of strict priority policies. Finally we compare the GPS and the GLQF policy in Section 6 and conclude are in Section 7.

2 Preliminaries

In this section we briefly review some basic results on the Large Deviations Theory [9, 15]. We first state the Gärtner-Ellis Theorem.

Consider a sequence $\{S_1, S_2, \dots\}$ of random variables, with values in $\mathbb{R}$ and define

$\Lambda_n(\theta) \stackrel{\Delta}{=} \frac{1}{n}\log \mathbf{E}[e^{\theta S_n}]$. For the applications that we have in mind, S_n is a partial sum process. Namely, $S_n = \sum_{i=1}^{n} X_i$, where X_i, $i \geq 1$, are identically distributed, possibly *dependent* random variables.

Assumption A

1. *The limit* $\Lambda(\theta) \stackrel{\Delta}{=} \lim_{n\to\infty} \Lambda_n(\theta) = \lim_{n\to\infty} \frac{1}{n}\log \mathbf{E}[e^{\theta S_n}]$, *exists for all* θ, *where* $\pm\infty$ *are allowed both as elements of the sequence* $\Lambda_n(\theta)$ *and as limit points.*

2. *The origin is in the interior of the domain* $D_\Lambda \stackrel{\Delta}{=} \{\theta \mid \Lambda(\theta) < \infty\}$ *of* $\Lambda(\theta)$.

3. $\Lambda(\theta)$ *is differentiable in the interior of* D_Λ *and the derivative tends to infinity as* θ *approaches the boundary of* D_Λ.

4. $\Lambda(\theta)$ *is lower semicontinuous, i.e.,* $\liminf_{\theta_n\to\theta} \Lambda(\theta_n) \geq \Lambda(\theta)$, *for all* θ.

Theorem 2.1 ***(Gärtner-Ellis)*** *Under Assumption A, the following inequalities hold*

Upper Bound: *For every closed set* F: $\limsup_{n\to\infty} \frac{1}{n}\log \mathbf{P}\left[\frac{S_n}{n} \in F\right] \leq -\inf_{a\in F} \Lambda^*(a)$.

Lower Bound: *For every open set* G: $\liminf_{n\to\infty} \frac{1}{n}\log \mathbf{P}\left[\frac{S_n}{n} \in G\right] \geq -\inf_{a\in G} \Lambda^*(a)$,

where $\Lambda^*(a) \stackrel{\Delta}{=} \sup_\theta(\theta a - \Lambda(\theta))$.

We say that $\{S_n\}$ satisfies a *Large Deviations Principle (LDP)* with *good rate function* $\Lambda^*(\cdot)$. $\Lambda(\cdot)$ and $\Lambda^*(\cdot)$ are convex duals.

A stronger concept than the LDP for the partial sum random *variable* $S_n \in \mathbb{R}$, is the LDP for the partial sum *process* (*Sample path LDP*)

$$S_n(t) = \frac{1}{n}\sum_{i=1}^{\lfloor nt \rfloor} X_i, \qquad t \in [0,1].$$

In a key paper [8], under certain mild mixing conditions on the stationary sequence $\{X_i;\ i \geq 1\}$, the authors establish an LDP for the process $S_n(\cdot)$ in $D[0,1]$. Their result is a starting point for our analysis in this paper. In particular, we will be assuming the following version of the sample path LDP.

Assumption B

For all $m \in \mathbf{N}$, *for every* $\epsilon_1, \epsilon_2 > 0$ *and for every scalars* $a_0, \ldots, a_{m-1}$, *there exists* $M > 0$ *such that for all* $n \geq M$ *and all* $k_0, \ldots, k_m$ *with* $1 = k_0 \leq k_1 \leq \cdots \leq k_m = n$,

$$e^{-\left(n\epsilon_2 + \sum_{i=0}^{m-1}(k_{i+1}-k_i)\Lambda^*(a_i)\right)} \leq \mathbf{P}[|S_{k_{i+1}} - S_{k_i} - (k_{i+1}-k_i)a_i| \leq \epsilon_1 n,\ i = 0, \ldots, m-1] \leq e^{\left(n\epsilon_2 - \sum_{i=0}^{m-1}(k_{i+1}-k_i)\Lambda^*(a_i)\right)}. \tag{1}$$

A detailed discussion of this Assumption, and the technical conditions under which it is satisfied can be found in [8]. In [8] it is proved that this assumption is satisfied by processes that are commonly used in modeling the input traffic to communication networks, that is, renewal processes, Markov modulated processes and correlated stationary processes with mild mixing conditions. The same set of processes satisfy the following [5], which can be viewed as the "convex dual analog" of Assumption B.

Assumption C

For all $m \in \mathbf{N}$ *there exists* $M > 0$ *and a function* $0 \leq \Gamma(y) < \infty$, *for all* $y > 0$, *such that for all* $n \geq M$ *and all* $k_0, \ldots, k_m$ *with* $1 = k_0 \leq k_1 \leq \cdots \leq k_m = n$,

$$\mathbf{E}[e^{\theta \cdot Z}] \leq \exp\{\sum_{j=1}^{m}[(k_j - k_{j-1})\Lambda(\theta_j) + \Gamma(\theta_j)]\}, \tag{2}$$

where $\theta = (\theta_1, \ldots, \theta_m)$ *and* $Z = (S_{k_0}, S_{k_2} - S_{k_1}, \ldots, S_{k_m} - S_{k_{m-1}})$.

On a notational remark, in the rest of the paper we will be denoting by $\Lambda_X(\cdot)$ and $\Lambda_X^*(\cdot)$ the limiting log-moment generating function and the large deviations rate function, respectively, of the process X.

3 The Multiclass Model and Scheduling Policies

Consider the system depicted in Figure 1. We assume a slotted time model (i.e., discrete

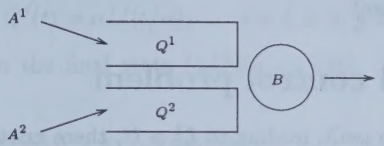


Figure 1: A multiclass model.

time) and we let A_i^1 (resp. A_i^2), $i \in \mathbb{Z}$, denote the number of type 1 (resp. 2) customers that enter queue Q^1 (resp. Q^2) at time i. Both queues have infinite buffers and share the same server which can process B_i customers during the time interval $[i, i+1]$. We assume that the processes $\{A_i^1;\ i \in \mathbb{Z}\}$, $\{A_i^2;\ i \in \mathbb{Z}\}$ and $\{B_i;\ i \in \mathbb{Z}\}$ are stationary and mutually independent. However, we allow dependencies between the number of customers at different slots in each process.

We denote by L_i^1 and L_i^2, the queue lengths at time i (without counting arrivals at time i) in queues Q^1 and Q^2, respectively. We assume that the server allocates its capacity between queues Q^1 and Q^2 according to a work-conserving policy, and that the queue length processes $\{L_i^j, j = 1, 2, i \in \mathbb{Z}\}$ are stationary.

For stability purposes we assume that for all i, $\mathbf{E}[B_i] > \mathbf{E}[A_i^1] + \mathbf{E}[A_i^2]$. We further assume that the arrival and service processes satisfy a LDP (Assumption A), as well as Assumptions B and C.

The GPS policy

According to the GPS policy the server allocates a fraction $\phi_1 \in [0, 1]$ of its capacity to queue Q^1, and the remaining fraction $\phi_2 = 1 - \phi_1$ to queue Q^2. The policy is defined to be work-conserving. (For a more formal definition see [14, 3]).

The GLQF policy

Consider the L^1-L^2 space and fix the parameter of the policy $\beta \geq 0$. There is a threshold line, of slope β, which divides the positive orthant of the $L^1 - L^2$ space in two regions. The GLQF policy serves Type 2 customers above the threshold line and Type 1 below it. The value $\beta = 1$ corresponds to the longest queue first (LQF) policy. (For a more formal definition see [14, 4]).

We are interested in estimating the overflow probability $\mathbf{P}[L_i^1 > U]$ for large values of U, at an arbitrary time slot i, in steady-state. Having determined this, the overflow probability of the second queue can be obtained by a symmetrical argument.

We will prove that the overflow probability satisfies $\mathbf{P}[L_i^1 > U] \sim e^{-U\theta^*_{GPS}}$, under the GPS policy, and $\mathbf{P}[L_i^1 > U] \sim e^{-U\theta^*_{GLQF}}$, under the GLQF policy, asymptotically, as $U \to \infty$. To this end, we will develop a lower bound on the overflow probability, along with a matching upper bound. Consider all scenarios (paths) that lead to an overflow under the GPS policy (resp. GLQF policy). We show that the probability of each such scenario ω_{GPS} (resp. ω_{GLQF}) asymptotically behaves as $e^{-U\theta_{GPS}(\omega_{GPS})}$ (resp. $e^{-U\theta_{GLQF}(\omega_{GLQF})}$), for some function $\theta_{GPS}(\omega_{GPS})$ (resp. $\theta_{GLQF}(\omega_{GLQF})$). For every ω_{GPS} (resp. ω_{GLQF}), this probability is a lower bound on $\mathbf{P}[L_i^1 > U]$, in the GPS (resp. GLQF) case. We select the

tightest lower bound by performing the minimization $\theta^*_{GPS} = \min_{\omega_{GPS}} \theta_{GPS}(\omega_{GPS})$ in the GPS case, and the minimization $\theta^*_{GLQF} = \min_{\omega_{GLQF}} \theta_{GLQF}(\omega_{GLQF})$ in the GLQF case. This amounts to solving a deterministic optimal control problem. Optimal trajectories (paths) of the control problem correspond to *most likely* overflow scenarios. We show that these must be of one out of two possible types. In other words, with high probability, overflow occurs in one out of two possible modes. We obtain an upper bound on $\mathbf{P}[L^1_i > U]$ by first obtaining a sample path upper bound, i.e., $L^1_i \leq \tilde{L}^1_i$ (where $\tilde{L}^1_i$ depends on the policy) and establishing that $\mathbf{P}[\tilde{L}^1_i > U]$ is at most $e^{-U\theta^*_{GPS}}$ in the GPS case (resp. $e^{-U\theta^*_{GLQF}}$ in the GLQF case).

4 The optimal control problem

For every overflow sample path, leading to $L^1_0 > U$, there exists some time $-n \leq 0$ that both queues are empty. Since we are interested in the asymptotics as $U \to \infty$, we scale time and the levels of the processes A^1, A^2 and B by U. We then let $T = \frac{n}{U}$ and define the following continuous-time functions in $D[-T, 0]$:

$$L^j(t) = \frac{1}{U} L^j_{\lfloor Ut \rfloor},\ j = 1, 2, \quad S^X(t) = \frac{1}{U} S^X_{-UT, \lfloor Ut \rfloor},\ X \in \{A^1, A^2, B\}, \quad \text{for } t \in [-T, 0].$$

Notice that the empirical rate of a process X is roughly equal to the rate of growth of $S^X(t)$. We let, $x_1(t), x_2(t)$ and $x_3(t)$ denote the empirical rates of the processes A^1, A^2 and B, respectively. The probability of sustaining rates $x_1(t), x_2(t)$ and $x_3(t)$, in the interval $[-UT, 0]$ for large values of U is given (up to first degree in the exponent) by

$$\exp\left\{-U \int_{-T}^{0} [\Lambda^*_{A^1}(x_1(t)) + \Lambda^*_{A^2}(x_2(t)) + \Lambda^*_B(x_3(t))]\, dt\right\}.$$

This cost functional is a consequence of Assumption B. With the scaling introduced here, as $U \to \infty$ the sequence of slopes $a_0, a_1, \ldots, a_{m-1}$ appearing there converges to the empirical rate $x(\cdot)$ and the sum of rate functions appearing in the exponent converges to an integral.

We seek a path with maximum probability, i.e., a minimum cost path where the cost functional is given by the integral in the above expression. This optimization is subject to the constraints $L^1(-T) = L^2(-T) = 0$ and $L^1(0) = 1$. The fluid levels in the two queues $L^1(t)$ and $L^2(t)$ are the state variables and the empirical rates $x_1(t), x_2(t)$ and $x_3(t)$ are the control variables. The dynamics of the system depend on the state and on the particular scheduling policy that is implemented. We refrain from explicitly describing here these dynamics due to lack of space. (See [14, 3, 4] for the complete picture). It is important though to note that both under GPS and GLQF, there are three regions of the state space with different set of dynamics in each region. Let (GPS-DYNAMICS) denote the set of state trajectories $L^j(t)$, $j = 1, 2$, $t \in [-T, 0]$, that obey the dynamics under the GPS policy, and (GLQF-DYNAMICS) the corresponding set under the GLQF policy. We next formally define the following optimal control problem (GPS-OVERFLOW), when the GPS policy is implemented.

$$\text{(GPS-OVERFLOW) minimize} \int_{-T}^{0} [\Lambda^*_{A^1}(x_1(t)) + \Lambda^*_{A^2}(x_2(t)) + \Lambda^*_B(x_3(t))]\, dt \tag{3}$$

$$\text{subject to: } L^1(-T) = L^2(-T) = 0$$

$$L^1(0) = 1, \quad L^2(0) : \text{ free}, \quad T : \text{ free}$$

$$\{L^j(t) :\ t \in [-T, 0],\ j = 1, 2\} \in \text{(GPS-DYNAMICS)}.$$

In the GLQF case, we substitute (GPS-DYNAMICS) with (GLQF-DYNAMICS) to obtain the corresponding control problem (GLQF-OVERFLOW).

The first property of these control problems that we show is that *optimal control trajectories can be taken to be constant* within each of the three regions. The result, which is based on convexity, can be found in [14, 3]. A trajectory is called optimal if it achieves the lowest cost among all trajectories with the same initial and final state. Since we have a free time problem, any segment of an optimal trajectory is also optimal.

Consider now a control trajectory $\{x_i^L(t);\ t \in [-T, 0]\}$ with corresponding state trajectory $\{L^1(t), L^2(t);\ t \in [-T, 0]\}$, which leads to a final state $(L^1(0), L^2(0))$. Define a scaled trajectory as

$$x_i^Q(t) = x_i^L(t/\alpha), \quad Q^j(t) = \alpha L^j(t/\alpha), \qquad i = 1, 2, 3,\ j = 1, 2,\ t \in [-\alpha T, 0],$$

and note that it leads to the final state $(\alpha L^1(0), \alpha L^2(0))$. Then, the cost of the Q trajectory is given by

$$\int_{-\alpha T}^{0} [\Lambda_{A^1}^*(x_1^Q(t)) + \Lambda_{A^2}^*(x_2^Q(t)) + \Lambda_B^*(x_3^Q(t))]\, dt =$$

$$\alpha \int_{-T}^{0} [\Lambda_{A^1}^*(x_1^L(t)) + \Lambda_{A^2}^*(x_2^L(t)) + \Lambda_B^*(x_3^L(t))]\, dt.$$

Using this observation, it follows easily that every scaled version of an optimal trajectory is optimal for the corresponding terminal state. We will call this *homogeneity* property of the control problems.

Optimal Solution of (GPS-OVERFLOW)

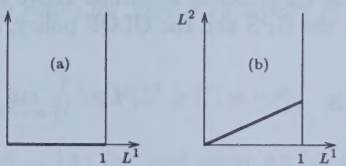


Figure 2: Optimal state trajectories for (GPS-OVERFLOW).

Using a rather complicated interchange argument (which can be found in [14, 3]), we can show that for every state trajectory of (GPS-OVERFLOW), there exists one of the form depicted in Figure 2(a) or (b) with no larger cost. Calculating the optimal value of these trajectories we obtain the following proposition:

Proposition 4.1 ***(GPS Lower Bound)*** *Assuming that the arrival and service processes satisfy Assumptions A and B, and under the GPS policy, the steady-state queue length L^1 satisfies*

$$\lim_{U \to \infty} \frac{1}{U} \log \mathbf{P}[L^1 > U] \geq -\theta_{GPS}^*, \tag{4}$$

where θ_{GPS}^* is given by $\theta_{GPS}^* = \min\left[\inf_{a>0} \frac{1}{a}\Lambda_{GPS}^{I*}(a), \inf_{a>0} \frac{1}{a}\Lambda_{GPS}^{II*}(a)\right]$, and the functions $\Lambda_{GPS}^{I*}(\cdot)$ and $\Lambda_{GPS}^{II*}(\cdot)$ are defined as follows: $\Lambda_{GPS}^{I*}(a) \triangleq \inf_{\substack{x_1+x_2-x_3=a \\ x_2 \leq \phi_2 x_3}} [\Lambda_{A^1}^*(x_1) + \Lambda_{A^2}^*(x_2) + \Lambda_B^*(x_3)]$, and $\Lambda_{GPS}^{II*}(a) \triangleq \inf_{\substack{x_1-\phi_1 x_3=a \\ x_2 \geq \phi_2 x_3}} [\Lambda_{A^1}^*(x_1) + \Lambda_{A^2}^*(x_2) + \Lambda_B^*(x_3)]$.

Optimal Solution of (GLQF-OVERFLOW)

We now turn our attention to the problem (GLQF-OVERFLOW). Again, by an interchange argument ([14, 4]), we can show that for every state trajectory of (GLQF-OVERFLOW), there exists one of the form depicted in Figure 3(a) or (b) with no larger cost. Evaluating the cost of the above trajectories we obtain the following proposition:

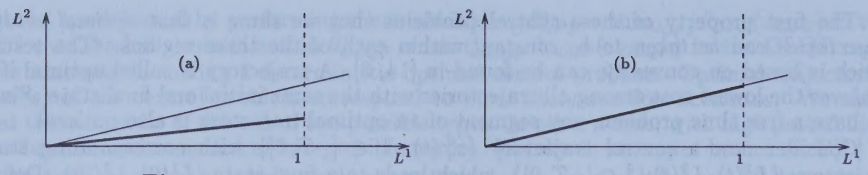


Figure 3: Optimal trajectories for (GLQF-OVERFLOW).

Proposition 4.2 ***(GLQF Lower Bound)*** *Assuming that the arrival and service processes satisfy Assumptions A and B, and under the GLQF policy, the steady-state queue length L^1 satisfies*

$$\lim_{U\to\infty} \frac{1}{U} \log \mathbf{P}[L^1 > U] \geq -\theta^*_{GLQF}, \tag{5}$$

where θ^*_{GLQF} *is given by* $\theta^*_{GLQF} = \min\left[\inf_{a>0} \frac{1}{a}\Lambda^{I*}_{GLQF}(a), \inf_{a>0} \frac{1}{a}\Lambda^{II*}_{GLQF}(a)\right]$, *and where we define:* $\Lambda^{I*}_{GLQF}(a) \triangleq \inf_{\substack{x_1-x_3=a \\ x_2\leq\beta(x_1-x_3)}} [\Lambda^*_{A^1}(x_1) + \Lambda^*_{A^2}(x_2) + \Lambda^*_B(x_3)]$, *and* $\Lambda^{II*}_{GLQF}(a) \triangleq \inf_{\substack{x_1-\phi x_3=a \\ x_2-(1-\phi)x_3=\beta a \\ 0\leq\phi<1}} [\Lambda^*_{A^1}(x_1) + \Lambda^*_{A^2}(x_2) + \Lambda^*_B(x_3)]$.

As we have explained in the Section 3, we have also established upper bounds that match the lower bounds in Propositions 4.1 and 4.2. This is sufficient to guarantee that the scenarios (optimal state trajectories) identified above are the *most likely ways* that queue Q^1 overflows, under the GPS and the GLQF policy, respectively.

5 Main Results

GPS policy

Theorem 5.1 ***(GPS Main)*** *Under the GPS policy, assuming that the arrival and service processes satisfy Assumptions A, B, and C the steady-state queue length, L^1 satisfies*

$$\lim_{U\to\infty} \frac{1}{U} \log \mathbf{P}[L^1 > U] = -\theta^*_{GPS}, \tag{6}$$

Proof : Proposition 4.1 provides the lower bound. The upper bound proof can be found in [14, 3]. ■

Next we state an alternative expression for θ^*_{GPS} (see [14, 3] for a proof), which may be more convenient in computations. Consider a convex function $f(u)$ with the property $f(0) = 0$. We define the *largest root* of $f(u)$ to be the solution of the optimization problem $\sup_{u:f(u)<0} u$. If $f(\cdot)$ has negative derivative at $u = 0$, there are two cases: either $f(\cdot)$ has a single positive root or it stays below the horizontal axis $u = 0$, for all $u > 0$. In the latter case, we will say that $f(\cdot)$ has a root at $u = \infty$.

Theorem 5.2 θ^*_{GPS} *is the largest positive root of the equation*

$$\Lambda_{GPS}(\theta) \triangleq \Lambda_{A^1}(\theta) + \inf_{0\leq u\leq\theta} [\Lambda_{A^2}(\theta - u) + \Lambda_B(-\theta + \phi_2 u)] = 0. \tag{7}$$

The GLQF policy

Similarly for the GLQF policy we have the following theorems. Due to limited space, the proofs are omitted and can be found in [14, 4].

Theorem 5.3 ***(GLQF Main)*** *Under the GLQF policy, assuming that the arrival and service processes satisfy Assumptions A, B, and C, the steady-state queue length, L^1 satisfies*

$$\lim_{U\to\infty} \frac{1}{U} \log \mathbf{P}[L^1 > U] = -\theta^*_{GLQF}, \tag{8}$$

Theorem 5.4 *θ^*_{GLQF} is the largest positive root of the equation*

$$\Lambda_{GLQF}(\theta) = \max\{\Lambda_{A^1}(\theta) + \Lambda_B(-\theta), \inf_{0\le u\le \frac{\theta}{1+\beta}} [\Lambda_{A^1}(\theta - u\beta) + \Lambda_{A^2}(u) + \Lambda_B(-u)]\} = 0. \tag{9}$$

Special Case: Priority Policies

Priority policies are a special case. Type 1 customers have higher priority when $\phi_1 = 1$ (or $\beta = \infty$) and lower priority when $\phi_1 = 0$ (or $\beta = 0$).

Corollary 5.5 ***(Priority policies)*** *Under strict priority policy for Type 1 customers (P_1), assuming that the arrival and service processes satisfy Assumptions A, B, and C the steady-state queue length L^1 satisfies*

$$\lim_{U\to\infty} \frac{1}{U} \log \mathbf{P}[L^1 > U] = -\theta^*_{P_1}, \tag{10}$$

where $\theta^*_{P_1}$ is given by $\theta^*_{P_1} = \inf_{a>0} \frac{1}{a}\Lambda^*_{P_1}(a)$, and where $\Lambda^*_{P_1}(a) \triangleq \inf_{x_1-x_3=a}[\Lambda^*_{A^1}(x_1) + \Lambda^*_B(x_3)]$. Under strict priority policy for Type 2 customers (P_2), the steady-state queue length L^1 satisfies

$$\lim_{U\to\infty} \frac{1}{U} \log \mathbf{P}[L^1 > U] = -\theta^*_{P_2}, \tag{11}$$

where $\theta^*_{P_2}$ is given by $\theta^*_{P_2} = \inf_{a>0} \frac{1}{a}\Lambda^*_{P_2}(a)$, and where $\Lambda^*_{P_2}(a) \triangleq \inf_{\substack{x_1+x_2-x_3=a \\ x_2\le x_3}}[\Lambda^*_{A^1}(x_1) + \Lambda^*_{A^2}(x_2) + \Lambda^*_B(x_3)]$.

Corollary 5.6 *θ^*_{P1} is the largest positive root of the equation*

$$\Lambda_{P1}(\theta) \triangleq \Lambda_{A^1}(\theta) + \Lambda_B(-\theta) = 0. \tag{12}$$

Also, θ^*_{P2} is the largest positive root of the equation

$$\Lambda_{P2}(\theta) \triangleq \Lambda_{A^1}(\theta) + \inf_{0\le u\le\theta}[\Lambda_{A^2}(\theta - u) + \Lambda_B(-\theta + u)] = 0. \tag{13}$$

6 Comparison

Let π be an arbitrary work-conserving policy that allocates the capacity of the server to the two queues Q^1 and Q^2, and Π the set of all work-conserving policies π. Let L^1 and L^2 denote the queue lengths of Q^1 and Q^2, respectively, at an arbitrary time slot, when the system operates under π. Let us now define θ^π the vector $(\theta^\pi_1, \theta^\pi_2)$ where

$$\theta^\pi_1 = \lim_{U\to\infty} \frac{1}{U} \log \mathbf{P}[L^1 > U] \qquad \text{and} \qquad \theta^\pi_2 = \lim_{U\to\infty} \frac{1}{U} \log \mathbf{P}[L^2 > U]. \tag{14}$$

The GPS policy is a parametric policy with performance depending on the parameter ϕ_1. To make this dependence explicit we will be using the notation GPS(ϕ_1). For the same reason we will be using GLQF(β). Using Theorems 5.1, 5.3 and Corollary 5.5 one can readily obtain the corresponding θ^π for the policies GPS(ϕ_1), GLQF(β), P_1 and P_2.

It is intuitively obvious that $\theta^{P_1} = (\max_{\pi\in\Pi}\theta_1^\pi, \min_{\pi\in\Pi}\theta_2^\pi)$ and $\theta^{P_2} = (\min_{\pi\in\Pi}\theta_1^\pi, \max_{\pi\in\Pi}\theta_2^\pi)$. In Figure 4 we plot $\theta^{GPS(\phi_1)}$ as ϕ_1 varies in $[0,1]$, and $\theta^{GLQF(\beta)}$ as β varies in $[0,\infty)$. For simplicity the calculations were performed with the arrival and service processes being Bernoulli. Note that $\theta^{P_1} = \theta^{GPS(1)} = \theta^{GLQF(\infty)}$ and that $\theta^{P_2} = \theta^{GPS(0)} = \theta^{GLQF(0)}$.

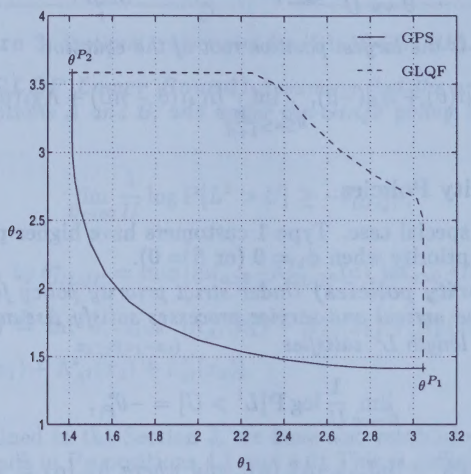


Figure 4: The performance $\theta^{GPS(\phi_1)}$ of the GPS(ϕ_1) policy as ϕ_1 varies in $[0,1]$, and the performance $\theta^{GLQF(\beta)}$ of the GLQF(β) policy as β varies in $[0,\infty)$, when A^1 ~Ber(0.3), A^2 ~Ber(0.2) and B ~Ber(0.9).

Figure 4 indicates that the GLQF policy achieves smaller overflow probabilities than the GPS. We establish that this is a general property. This suggests than when one has to deal with delay insensitive traffic GLQF is more suitable than GPS. Let us first formally define the term *the GLQF curve dominates the GPS curve.*

Definition 6.1
We say that the GLQF curve dominates the GPS curve *when there does not exist a pair of* $\phi_1 \in [0,1]$ *and* $\beta \in [0,\infty)$ *satisfying* $\theta_1^{GPS(\phi_1)} > \theta_1^{GLQF(\beta)}$ *and* $\theta_2^{GPS(\phi_1)} > \theta_2^{GLQF(\beta)}$.

We next state (see [14, 4] for a proof) a sufficient condition for the GLQF curve to dominate the GPS curve.

Lemma 6.2 *If for all* $\beta \in [0,\infty)$ *there exists* $\phi_1 \in [0,1)$ *such that*

$$\theta_1^{GPS(\phi_1)} \le \theta_1^{GLQF(\beta)} \qquad \text{and} \qquad \theta_2^{GPS(\phi_1)} \le \theta_2^{GLQF(\beta)},$$

then the GLQF curve dominates the GPS curve.

Our main result in this section is the following (see [14, 4] for a proof):

Theorem 6.3 *Assuming that the arrival and service processes satisfy Assumptions A, C, and B, the GLQF curve dominates the GPS curve.*

7 Conclusions

In this paper we considered a multiclass multiplexer, with segregated buffers for each type of traffic, and under the GPS and the GLQF policy we have obtained the asymptotic (as the buffer size goes to infinity) tail of the overflow probability for each buffer. We have explicitly and in detail characterized the most likely modes of overflow. We formulated the problem of calculating the maximum overflow probability (over all scenarios that

lead to overflow) as an optimal control problem, general enough to include any work conserving policy. This provides particular insight into the problem. The results can be used for admission control purposes in multiclass switches (see [14]).

References

[1] D. Bertsimas, I. C. Paschalidis, and J. N. Tsitsiklis. On the large deviations behaviour of acyclic networks of G/G/1 queues. Tech. Rep. LIDS-P-2278, LIDS, MIT, December 1994.

[2] D. Bertsimas, I. C. Paschalidis, and J. N. Tsitsiklis. On the large deviations behaviour of acyclic single class networks and multiclass queues. Talk at the RSS Workshop in Stochastic Networks, Edinburgh, U.K., 1995.

[3] D. Bertsimas, I. C. Paschalidis, and J. N. Tsitsiklis. Asymptotic buffer overflow probabilities in multiclass multiplexers, Part I: The GPS policy. Tech. Rep. LIDS-P-2341, LIDS, MIT, June 1996.

[4] D. Bertsimas, I. C. Paschalidis, and J. N. Tsitsiklis. Asymptotic buffer overflow probabilities in multiclass multiplexers, Part II: The GLQF policy. Tech. Rep. LIDS-P-2342, LIDS, MIT, June 1996.

[5] C. Chang. Sample path large deviations and intree networks. *Queueing Systems*, 20:7–36, 1995.

[6] C. Courcoubetis and R. Weber. Estimation of overflow probabilities for state-dependent service of traffic streams with dedicated buffers. Talk at the RSS Workshop in Stochastic Networks, Edinburgh, U.K., 1995.

[7] G. de Veciana and G. Kesidis. Bandwidth allocation for multiple qualities of service using Generalized Processor Sharing. Tech. Rep. SCC-94-01, Dept. of ECE, The U. of Texas at Austin, 1994; Revised 1995.

[8] A. Dembo and T. Zajic. Large deviations: From empirical mean and measure to partial sums processes. Preprint, 1993.

[9] A. Dembo and O. Zeitouni. *Large Deviations Techniques and Applications.* Jones and Bartlett, 1993.

[10] A. I. Elwalid and D. Mitra. Analysis, approximations and admission control of a multiple-service multiplexing system with priorities. Preprint, 1994.

[11] H. Gail, G. Grover, R. Guérin, S. Hantler, Z. Rosberg, and M. Sidi. Buffer size requirements under longest queue first. *Performance Evaluation*, 18:133–140, 1993.

[12] N. O'Connell. Queue lengths and departures at single-server resources. Talk at the RSS Workshop in Stochastic Networks, Edinburgh, U.K., 1995.

[13] A. Parekh and R. Gallager. A generalized processor sharing approach to flow control in integrated services networks: The single node case. *IEEE/ACM Transactions on Networking*, 1(3):344–357, 1993.

[14] I. C. Paschalidis. *Large Deviations in High Speed Communication Networks.* PhD thesis, MIT, May 1996.

[15] A. Shwartz and A. Weiss. *Large deviations for performance analysis.* Chapman and Hall, New York, 1995.

[16] Z.-L. Zhang. Large deviations and the generalized processor sharing scheduling: Upper and lower bounds. Part I:Two-queue systems. Tech. Rep., CS Dept., U. of Massachusetts at Amherst, 1995.

Functional Large Deviation Principles for Queues

Anatolii A. Puhalskii
Institute for Problems
in Information Transmission
19 Bolshoi Karetnii
Moscow 101447 Russia
pukh@ippi.ac.msk.su

Ward Whitt
AT&T Labs
Room 2C-178
Murray Hill, NJ 07974-0636
wow@research.att.com

Abstract

We establish functional large deviation principles (FLDPs) for waiting times and departure times in single-server queues with unlimited waiting space. We apply the extended contraction principle to show that these processes obey FLDPs in the function space D whenever the arrival and service processes obey FLDPs. However, the rate function for the departure process is complicated for non-deterministic service times. We exhibit the rate function in the case of deterministic interarrival times. We also establish an FLDP for inverse processes, which we apply to establish FLDPs for the arrival counting process and the arrival times in a superposition of independent arrival processes. We combine the results to obtain an FLDP for waiting times in the $\sum_{i=1}^{k} GI_i/GI/1$ queue.

1. Introduction

Just as functional central limit theorems (FCLTs) are useful to establish ordinary central limit theorems for various functionals of stochastic processes [2], [19], so are *functional* (or sample path) *large deviation principles* (FLDPs) useful to establish ordinary large deviations principles (LDPs) for various functionals of stochastic processes [11]. The contraction principle and its extensions play the role for FLDPs that the continuous mapping theorem and its extensions play for FCLTs.

In this paper we briefly describe FLDPs for inverse (or first passage time) processes and queueing processes that we have obtained in [14], [15]. These papers extend and complement our previous work in [7], [8], [13] as well as the work of others, e.g., [1], [3], [17]. The FLDPs for inverse processes enable us to obtain here an FLDP for the waiting times in the $\sum_{i=1}^{k} GI_i/GI/1$ queue, which has an arrival process that is a superposition of renewal processes (Theorem 5.2). The inverse FLDPs also play a role in establishing FLDPs for split point processes, which arise when departures from one queue are routed to one of several other queues or leave the network.

Our goal is to establish FLDPs for waiting times at each queue in an acyclic network of single-server queues by showing how FLDPs are inherited under the network operations of departure, splitting and superposition. The resulting LDPs for waiting times then may be applied to yield effective bandwidths in communication networks [4], [18]. In an acyclic network, we should be able to recursively determine each successive rate function governing the FLDP for the waiting times at each successive queue. However, determining the rate function for the departure process requires the solution of a difficult optimization problem, which evidently must be done numerically. For special cases it simplifies though. For the special case of deterministic service times, the departure rate function reduces to a minor modification of the arrival rate function, which is consistent with previous results by de Veciana, Courcoubetis and Walrand [17] and Chang [3].

A major feature of our work compared to previous LDP work such as in [3], [17] is working with the function space D endowed with the Skorohod [16] topologies. This framework is important for treating continuous-time processes with discontinuous sample paths. Even the ordinary Poisson process is not a bonafide random element of D with the Borel σ-field generated by the topology of uniform convergence on compact subintervals [2], Section 18.

All the LDPs and FLDPs in this paper have the usual linear scaling. For treating queues with long-range-dependent input, it can be important to consider nonlinear scaling, as was done in [5]. Some FLDPs with nonlinear scaling are established in [6].

2. Functions and Topologies on D and its Subsets

In this section we provide the theoretical basis for establishing FLDPs for the sequence of successive arrival times in a superposition of independent arrival processes, as needed to treat statistical multiplexing. We start with FLDPs for the component arrival times; we apply the inverse map to get FLDPs for the associated arrival counting processes; we add to get an FLDP for the aggregate superposition counting process; and we apply the inverse map once again to get a FLDP for the arrival times of the superposition process. This program parallels previous heavy-traffic FCLTs for queueing networks [9], [19].

Let D be the space of all right-continuous real-valued functions $x \equiv (x(t),\ t \geq 0)$ on the nonnegative halfline $[0,\infty)$ with left limits everywhere in $(0,\infty)$. Let E be the subset of functions x in D that are unbounded above and satisfy $x(0) \geq 0$. Let $D^{\uparrow}$ be the subset of nondecreasing functions in D and let $E^{\uparrow} = E \cap D^{\uparrow}$.

Let the *inverse function* be defined for any $x \in E$ by

$$x^{-1}(t) \equiv \inf\{s > 0 : x(s) > t\}\ , \quad t \geq 0\ . \tag{2.1}$$

Obviously, if $x \in E$, then $x^{-1} \in E^{\uparrow}$.

This section largely follows Section 7 of Whitt [19], but there will be a correction for treating the inverse function with the M_1 topology. (This correction is relevant for FCLTs as well as for FLDPs.) We consider the Skorohod [16] J_1 and M_1 topologies on D, with the domain extended from $[0,1]$ to $[0,\infty)$, and a minor modification of M_1 topology denoted by M_1'. For these topologies, D is metrizable as a separable metric space. Let D have the Borel σ-field induced by its topology. For the topologies we consider, the Borel σ-fields coincide with the usual Kolmogorov σ-field generated by the finite-dimensional projection maps. The M_1 topology on D is defined in terms of the *completed graph*

$$\Gamma(x) \equiv \{(u,t) \in R \times R_+ \backslash \{0\} : u \in [x(t-) \wedge x(t)\ ,\ \ x(t-) \vee x(t)]\} \cup \{(x(0),0)\}\ , \tag{2.2}$$

where $x(t-)$ denotes the left limit of x at t, $\wedge$ denotes the minimum and $\vee$ denotes the maximum. We will call a pair of continuous functions $(u,t) \equiv ((u(s),t(s)), s \geq 0)$ such that the $t(s)$ is nondecreasing with $t(0) = 0$ a *parametrization* of $\Gamma(x)$ if $\Gamma(x) = \cup\{(u(s),t(s)) : s \geq 0\}$. A sequence $\{x_n,\ n \geq 1\}$ in D converges to x in $D(M_1)$ if there exist parametrizations (u_n,t_n) of $x_n, n \geq 1$, and (u,t) of x such that

$$\sup_{s \leq T}\{|u_n(s) - u(s)| + |t_n(s) - t(s)|\} \to 0 \ \text{ as } \ n \to \infty \text{ for all } T > 0\ . \tag{2.3}$$

What we would like is for the inverse function in (2.1) to be continuous on E, but we must impose constraints when we work with the J_1 and M_1 topologies.

Lemma 2.1. *The inverse function in (2.1) is measurable on E, is continuous in the M_1 topology at those x for which $x^{-1}(0) = 0$, and is continuous in the J_1 topology at each strictly increasing x.*

To see that the M_1 condition is needed, let $x_n(t) = t/n$, $0 \le t < 1$, and $x_n(t) = t - 1$, $t \ge 1$. Clearly $x_n \to x$ (M_1) where $x(t) = 0$, $0 \le t < 1$ and $x(t) = t - 1$, $t \ge 1$. However, $x_n^{-1}(0) = 0 \not\to x^{-1}(0) = 1$ as $n \to \infty$, so that $x_n^{-1} \not\to x^{-1}(M_1)$ as $n \to \infty$.

Another approach to the problem of the continuity of the inverse mapping on D is to change the topology instead of adding the extra condition. Here we use a topology called M_1' that is defined in the same way as M_1, except that we change $\Gamma(x)$ to

$$\Gamma'(x) \equiv \{(u,t) \in R \times R_+ : u \in [x(t-) \wedge x(t), x(t-) \vee x(t)]\} , \tag{2.4}$$

where $x(0-) = 0$. Stated another way, $\Gamma'(x)$ is the extended graph $\Gamma(x)$ supplemented by adding the segment $[0, x(0)]$ if $x(0) \ge 0$ or $[x(0), 0]$ if $x(0) \le 0$. We say that $x_n \to x$ in $D(M_1')$ if (2.3) holds for some parametrizations of $\Gamma'(x_n)$ and $\Gamma'(x)$. It is not difficult to show that the M_1' topology is metrizable as a separable metric space and that it induces the Kolmogorov σ-field. In addition, the topologies are ordered by $J_1 \to M_1 \to M_1'$. Convergence $x_n \to x$ is equivalent for all three topologies at continuous x with $x(0) = 0$. Moreover, on $E^{\uparrow}(M_1')$ $x_n \to x$ is equivalent to pointwise convergence $x_n(t) \to x(t)$ at all continuity points except possibly for $t = 0$. The following is the key lemma.

Lemma 2.2. *The inverse function in (1.1) on E is continuous in the M_1' topology.*

Proof. The claim for the inverse function follows since if (u,t) is a Γ'-parametrization of $x \in E^{\uparrow}$, then (t,u) is directly a Γ'-parametrization of x^{-1}. □

3. Initial Large Deviation Conclusions

We now draw large deviation conclusions from the continuity properties in Section 2. We say that a function $I(x)$ defined on a metric space S and taking values in $[0, \infty]$ is a *rate function* if the sets $\{x \in S : I(x) \le a\}$ are compact for all $a \ge 0$, and a sequence $\{P_n, n \ge 1\}$ of probability measures on the Borel σ-field of S (or a sequence of random elements $\{X_n, n \ge 1\}$ with values in S and distributions P_n) obeys the LDP with the rate function I if

$$\overline{\lim_{n\to\infty}} \frac{1}{n} \log P_n(F) \le - \inf_{x \in F} I(x)$$

for all closed $F \subset S$, and

$$\underline{\lim_{n\to\infty}} \frac{1}{n} \log P_n(G) \ge - \inf_{x \in G} I(x)$$

for all open $G \subset S$. If the LDP is in a space of stochastic process sample paths such as D, then we call it an FLDP.

We establish new LDPs from previously established ones by applying the contraction principle or an extension [11], [13], Section 2. The *contraction principle* states that if $\{X_n, n \ge 1\}$ obeys an FLDP with rate function I and if f is continuous, then $\{f(X_n), n \ge 1\}$ obeys an LDP with rate function

$$I'(y) \equiv \inf_{x : f(x) = y} I(x) . \tag{3.1}$$

The *extended contraction principle* states that if $\{X_n, n \ge 1\}$ obeys an LDP with rate function I, if $\{f_n, n \ge 1\}$ is a sequence of measurable functions, if function f is continuous

in restriction to the sets $\{x : I(x) \le a\}, a \ge 0$, and if $f_n(x_n) \to f(x)$ as $n \to \infty$ for all x_n for which $x_n \to x$ as $n \to \infty$ for all x for which $I(x) < \infty$, then $\{f_n(X_n), n \ge 1\}$ obeys an LDP with rate function (3.1). An important special case is $f_n = f$, as in the contraction principle, where f is continuous at each x with $I(x) < \infty$. In either case, if in addition f is a bijection, then we can write $I'(y) = I(f^{-1}(y))$. The applications here illustrate the importance of the extended contraction principle. Let $x^{\uparrow}(t) = \sup\{x(s) : 0 \le s \le t\}$.

Theorem 3.1. *(a) If $\{X_n, n \ge 1\}$ obeys an FLDP in $E(J_1)$ with rate function I_X and $I_X(x) = \infty$ whenever x is not strictly increasing, then $\{X_n^{-1},\ n \ge 1\}$ obeys an FLDP in $E^{\uparrow}(J_1)$ with rate function*

$$I_{X^{-1}}(x) \equiv \inf_{\substack{y \in E \\ y \cdot y^{-1} = x}} \{I_X(y)\} = I_{X^{\uparrow}}(x^{-1}) \equiv \inf_{\substack{y \in E: \\ x = y^{\uparrow}}} I_X(y)\ ,\ x \in E^{\uparrow}\ . \tag{3.2}$$

(As a consequence, $I_{X^{-1}}(x) = \infty$ if x is not continuous.)
(b) If $\{X_n, n \ge 1\}$ obeys an FLDP in $E(M_1)$ with rate function I_X and if $I_X(x) = \infty$ whenever $x^{-1}(0) > 0$, then $\{X_n^{-1},\ n \ge 1\}$ obeys an FLDP in $E^{\uparrow}(M_1)$ with rate function $I_{X^{-1}}$ in (3.2). (As a consequence $I_{X^{-1}}(x) = \infty$ if $x(0) > 0$).
(c) If $\{X_n,\ n \ge 1\}$ obeys an FLDP in $E(M_1')$ with rate function I_X, then $\{X_n^{-1},\ n \ge 1\}$ obeys an FLDP in $E^{\uparrow}(M_1')$ with rate function $I_{X^{-1}}$ in (3.2).

It may be convenient to establish an FLDP for an inverse process by applying Theorem 3.1(b) or (c) for the M_1 or M_1' topology instead of the stronger J_1 topology, but the LDP extends to the stronger J_1 topology from M_1 or M_1' if the rate function $I_{X^{-1}}(x)$ is infinite at discontinuous x and for M_1' if in addition the rate function $I_{X^{-1}}(x) = \infty$ when $x(0) \ne 0$. This is because convergence $x_n \to x$ for continuous x with $x(0) = 0$ is equivalent for the three topologies. Indeed, the FLDP extends to the stronger uniform topology, under which D is nonseparable, provided that X_n^{-1} remains a bonafide random element. However, in general this need not be the case; see [2], Section 18. Indeed, measurability with respect to the Borel σ-field associated with the uniform topology fails even for the Poisson process. Thus, for general FLDPs on D it is often important to work with topologies like the Skorohod topologies.

In many (but not all) cases the rate functions $I_X(x)$ for X_n and $I_{X^{-1}}(x)$ for X_n^{-1} have the form

$$I_X(x) = \int_0^{\infty} \lambda_X(\dot{x}(t))dt\ ,\quad I_{X^{-1}}(x) = \int_0^{\infty} \lambda_{X^{-1}}(\dot{x}(t))dt \tag{3.3}$$

if x is absolutely continuous with $x(0) = 0$ and $I_X(x) = \infty$ and $I_{X^{-1}}(x) = \infty$ otherwise, where λ_X and $\lambda_{X^{-1}}$ are convex local rate functions on $\mathbb{R}$. We can then apply Theorem 3.1 to deduce the relation between the rate functions λ_X and $\lambda_{X^{-1}}$ on $\mathbb{R}$, which is consistent with what was established directly in [8].

Theorem 3.2. *If $\{X_n, n \ge 1\}$ obeys an FLDP in $E^{\uparrow}$ for one of the topologies J_1, M_1 or M_1' with the rate function I_X satisfying (3.3), where $\lambda_X(0) = \infty$, then $\{X_n^{-1}, n \ge 1\}$ obeys an FLDP in $E^{\uparrow}(J_1)$ with the rate function $I_{X^{-1}}$ from (3.3), where*

$$\lambda_{X^{-1}}(z) = z\lambda_X(1/z), \quad \lambda_{X^{-1}}(0) = \infty. \tag{3.4}$$

If the function λ_X (respectively, $\lambda_{X^{-1}}$) is convex downwards, then the sequence of random variables $\{X_n(1), n \ge 1\}$ (respectively, $\{X_n^{-1}(1), n \ge 1\}$) obeys the LDP in $\mathbb{R}$ with rate function λ_X (respectively, $\lambda_{X^{-1}}$).

It is important to note, however, that the rate functions I_X of $\{X_n, n \ge 1\}$ and $I_{X^{-1}}$ of $\{X_n^{-1}, n \ge 1\}$ may involve functions with jumps, so that (3.3) need not hold [10], e.g., this can occur with a renewal process [13], [14]. Then the connections to LDPs on

$\mathbb{R}$ is more complicated; e.g., because the projection map is not necessarily continuous. Functions with jumps may play a role in either I_X or $I_{X^{-1}}$ or both.

4. Superpositions of Renewal Processes

In this section, we apply the results of previous sections to derive FLDPs for the counting process and arrival times in a superposition of arrival processes. We assume that the component rate functions are finite only for continuous functions x; otherwise, we have the difficulty that addition is not continuous on $D \times D$ [19]. We specifically consider superpositions of renewal processes, for which the conditions can be expressed directly in terms of the interarrival time distributions. However, similar results hold for more general arrival processes provided that the conditions of Theorems 3.1 and 3.2 hold. For such more general results, see Chang [3] and Corollary 6.6 in Puhalskii [12].

Let $\{\xi_i^j, i \geq 1\}$, $1 \leq j \leq k$, be k independent sequences of i.i.d. nonnegative random variables with $E\xi_1^j > 0$. Let $(N^j(t), t \geq 0)$, $1 \leq j \leq k$, be the associated k mutually independent renewal counting processes and let $N = N^1 + \ldots + N^k$. For each j, let X_n^j be the normalized partial sum process and let X_n be the normalized partial sum process associated with the superposition process, defined by

$$X_n^j(t) \equiv n^{-1} \sum_{i=1}^{\lfloor nt \rfloor} \xi_i^j, \quad X_n(t) \equiv n^{-1} \sum_{i=1}^{\lfloor nt \rfloor} \xi_i \ , \quad t \geq 0 \ , \tag{4.1}$$

where ξ_i is the i^{th} interval between points in the superposition process N. Let N_n^j and N_n be associated normalized counting processes, i.e.,

$$N_n^j(t) \equiv n^{-1} N^j(nt) \text{ and } N_n(t) \equiv n^{-1} N(nt) \ , \quad t \geq 0 \ . \tag{4.2}$$

We will derive LDPs for $\{N_n, n \geq 1\}$ and $\{X_n, n \geq 1\}$. For this purpose, note that $N_n^j(t) = (X_n^j)^{-1}(t) - n^{-1}$, so that

$$N_n^j = (X_n^j)^{-1} - n^{-1}, \ N_n = X_n^{-1} - n^{-1} \ , \quad X_n = (N_n + n^{-1})^{-1} \ . \tag{4.3}$$

Theorem 4.1. *Assume that $E\exp(\alpha\xi_1^j) < \infty$, $1 \leq j \leq k$, for some $\alpha > 0$. Let $\alpha_j^* \equiv \sup\{\alpha : E\exp(\alpha\xi_1^j) < \infty\}, 1 \leq j \leq k$, and $\alpha^* = \sum_{j=1}^k \alpha_j^*$. Also assume that $P(\xi_1^j = 0) = 0$ for all j, $1 \leq j \leq k$. Then the sequence $\{N_n, n \geq 1\}$ in (4.2) obeys an FLDP in $E^\uparrow$ for the J_1 topology with rate function*

$$I_N(x) = \int_0^\infty \sup_{\alpha < \alpha^*} \{\alpha - \dot{x}(t)\psi(\alpha)\} dt = \int_0^\infty \sup_{\alpha < \overline{\alpha}} \{\phi(\alpha) - \alpha\dot{x}(t)\} dt \ , \tag{4.4}$$

where

$$\psi(\alpha) = \phi^{-1}(\alpha) \ , \quad \phi(\alpha) = \sum_{j=1}^k \phi_j(\alpha) \ , \quad \phi_j(\alpha) = \psi_j^{-1}(\alpha) \ , \tag{4.5}$$

$$\overline{\alpha} = \min_{1 \leq j \leq k} \lim_{\alpha \uparrow \alpha_j^*} \psi_j(\alpha) \text{ with } \psi_j(\alpha) = \log E\exp(\alpha\xi_1^j) \ , \tag{4.6}$$

if x is absolutely continuous and $x(0) = 0$, while $I_N(x) = \infty$ otherwise. If in addition there is one j for which $E\exp(\alpha\xi_1^j) < \infty$ for all α, then $\{X_n, n \geq 1\}$ in (4.1) obeys an FLDP in $E^\uparrow$ for the J_1 topology with rate function

$$I_X(x) = I_N(x^{-1}) = \int_0^\infty \sup_{\alpha \in R} \{\alpha\dot{x}(t) - \psi(\alpha)\} dt = \int_0^\infty \sup_{\alpha \in R} \{\dot{x}(t)\phi(\alpha) - \alpha\} dt \ , \tag{4.7}$$

if x is absolutely continuous with $x(0) = 0$, and $I_X(x) = \infty$ otherwise.

5. Large Deviations for Waiting Times

We now state FLDPs that we have established for the waiting times in a single server queue with unlimited waiting room. Let v_n be the service time of the n^{th} customer and let u_n be the interarrival time between customers n and $(n+1)$. Let a 0^{th} customer arrive at time 0 to find an empty system. Then the waiting time of the n^{th} customer satisfies

$$w_n = [w_{n-1} + x_{n-1}]^+ = s_n - \min\{s_k : 0 \le k \le n\} ,$$

where $x_n = v_n - u_n$, $s_n = x_0 + \ldots + x_{n-1}$ and $s_0 = 0$. Hence, introducing the processes

$$U_n(t) = n^{-1} \sum_{i=1}^{\lfloor nt \rfloor} u_{i-1}, \quad V_n(t) = n^{-1} \sum_{i=1}^{\lfloor nt \rfloor} v_{i-1} , \quad W_n(t) = n^{-1} w_{\lfloor nt \rfloor}, \ t \ge 0 , \tag{5.1}$$

we have

$$W_n = R(V_n - U_n), \ n \ge 1 , \tag{5.2}$$

where R is the one-dimensional reflection map on D, defined by

$$R(x)(t) = x(t) - \inf\{x(s) : 0 \le s \le t\} \wedge 0, \ t \ge 0 . \tag{5.3}$$

The following is a version of Lemma 4.6 of [13].

Lemma 5.1. *Let z be nonnegative and x be absolutely continuous. Then $z = R(x)$ where R is the reflection map in (5.3) if and only if z is absolutely continuous, $\dot{z}(t) = \dot{x}(t)$ a.e. on the set $\{z(t) > 0\}$ and $\dot{x}(t) \le 0$ a.e. on the set $\{z(t) = 0\}$.*

Let 1_A be the indicator function of the set A, which is 1 on A and 0 elsewhere.

Theorem 5.1. *(a) if $\{(U_n, V_n), n \ge 1\}$ for U_n and $\{V_n\}$ in (5.1) obeys an FLDP in $D \times D$ for (the product topology associated with) one of the topologies J_1, M_1 or M_1' with rate function $I_{U,V}$ where $I_{U,V}(u,v) = \infty$ if both u and v are discontinuous, then $\{W_n, n \ge 1\}$ in (5.1) obeys an FLDP in D for the same topology with rate function*

$$I_W(w) = \inf_{\substack{u,v \in D \times D: \\ w = R(v-u)}} \{I_{U,V}(u,v)\} . \tag{5.4}$$

(b) In addition, suppose that $I_{U,V}(u,v) = I_U(u) + I_V(v)$, where I_U and I_V are integrals of local rate functions λ_U and λ_V as in (3.3), then for nonnegative absolutely continuous w with $w(0) = 0$

$$\begin{aligned} I_W(w) = & \int_0^\infty 1_{\{w(t)>0\}} \inf_{0 \vee (-\dot{w}(t)) \le z} \{\lambda_U(z) + \lambda_V(\dot{w}(t) + z)\} dt \\ & + \inf_{0 \le y \le z} \{\lambda_U(z) + \lambda_V(y)\} \int_0^\infty 1_{\{w(t)=0\}} dt , \end{aligned} \tag{5.5}$$

while $I_W(w) = \infty$ otherwise.

(c) If, in addition,

$$\lambda_U(z) = \sup_{\alpha \in \mathbb{R}} \{\alpha z - \psi_U(\alpha)\} \quad \text{and} \quad \lambda_V(z) = \sup_{\alpha \in \mathbb{R}} \{\alpha z - \psi_V(\alpha)\}, \tag{5.6}$$

where $\psi_U(\alpha)$ and $\psi_V(\alpha)$ are convex, nondecreasing finite real-valued functions with $\psi_U(0) = \psi_V(0) = 0$, then for nonnegative absolutely continuous w with $w(0) = 0$

$$\begin{aligned} I_W(w) = & \int_0^\infty 1_{\{w(t)>0\}} \sup_{\beta \in \mathbb{R}} \{\beta \dot{w}(t) - \psi_U(-\beta) - \psi_V(\beta)\} dt \\ & + 1_{\{\rho > 1\}} \sup_{\beta \le 0} \{-\psi_U(-\beta) - \psi_V(\beta)\} \int_0^\infty 1_{\{w(t)=0\}} dt , \end{aligned} \tag{5.7}$$

where $\rho = \dot{\psi}_V(0)/\dot{\psi}_U(0)$ *and* $I_W(w) = \infty$ *otherwise.*

The natural sufficient condition for $\{(U_n, V_n), n \geq 1\}$ to obey an FLDP with rate function $I_{U,V}(u,v) = I_U(u) + I_V(v)$ is for $\{U_n, n \geq 1\}$ and $\{V_n, n \geq 1\}$ to be independent and separately obey FLDPs with rate functions $I_U(u)$ and $I_V(v)$. We will show in Section 6 below that the rate function of the departure process has the form required for the arrival process in part (a) but not in parts (b) and (c) of Theorem 5.1.

Since the rate function I_W in (5.7) is infinite at discontinuous arguments, we can apply the extended contraction principle with the projection map to obtain an LDP for the sequences $\{w_{\lfloor nt \rfloor}/n,\ n \geq 1\}$ in $\mathbb{R}$. For sufficiently large t, we can directly read off (but not rigorously prove) an LDP for the steady-state waiting time distribution.

Corollary 1. *If, in addition to the assumptions of Theorem 5.1 (b),* λ_V *and* λ_U *are convex functions with* $\lambda_V(v) = \lambda_U(u) = 0$ *for some points* u *and* v *with* $0 \leq v \leq u$ *(as anticipated for* $\rho \leq 1$*), then for each* $t > 0$ $\{w_{\lfloor nt \rfloor}/n,\ n \geq 1\}$ *obeys an LDP in* $\mathbb{R}$ *with rate function*

$$I_{w(t)}(z) = \inf_{\substack{x \in D \\ z = x(t)}} \{I_W(x)\} = z \left(\inf_{y \geq z/t} \{\lambda_X(y)/y\} \right) , \tag{5.8}$$

where

$$\lambda_X(z) = \inf_{y \geq 0} \{\lambda_U(y) + \lambda_V(y+z)\} . \tag{5.9}$$

If $\lambda_X(0) > 0$*, then, for all* t *sufficiently large,*

$$I_{w(t)}(z) = z \left(\inf_{y \geq 0} \{\lambda_X(y)/y\} \right) . \tag{5.10}$$

Note that (5.9) simplifies to $\lambda_X(z) = \lambda_U(v - z)$ if the service times are deterministic with value v, and to $\lambda_X(z) = \lambda_V(u + z)$ if the interarrival times are deterministic with value u.

Corollary 2. *Under the assumptions of Theorem 5.1 (c), (5.9) becomes*

$$\lambda_X(z) = \sup_{\beta \in \mathbb{R}} \{\beta z - \psi_U(-\beta) - \psi_V(\beta)\} .$$

Assume that $\psi_U(-\beta) + \psi_V(\beta) < 0$ *for some* β*. Then, for* t *sufficiently large,*

$$I_{w(t)}(z) = zx^*,$$

where $x^* = \sup\{\beta : \psi_U(-\beta) + \psi_V(\beta) \leq 0\}$.

The large t result in Corollary 2 is consistent with the result for the steady-state waiting time in Glynn and Whitt [7].

We now apply Theorems 4.1 and 5.1 to treat the $\sum_{i=1}^k GI_i/GI/1$ queue which has i.i.d. service times independent of an arrival process that is the superposition of independent renewal processes. Let ξ_j^i be the j^{th} interarrival time in component arrival process i.

Theorem 5.2. *In the* $\sum_{i=1}^k GI_i/GI/1$ *queue, the conclusions of Theorem 5.1 hold for the* J_1 *topology if* $(i) E\exp(\alpha v_1) < \infty$ *for all* α*,* (ii) $P(\xi_1^i > 0) = 1$ *for all* i*,* $1 \leq i \leq k$*, and* (iii) $E\exp(\alpha \xi_1^i) < \infty$ *for some* $\alpha > 0$ *for all* i*,* $1 \leq i \leq k$*, and for at least one* i*,* $E\exp(\alpha \xi_1^i) < \infty$ *for all* α*. Then* $\psi_V(\alpha) = \log E\exp(\alpha v_1)$*,* $\psi_U(\alpha)$ *is given in (4.5) and* ρ *is the traffic intensity.*

Theorem 5.2 extends to the more general $\sum_{i=1}^k G_i/G/1$ model where the arrival process is the superposition of k independent general arrival processes, which are independent of a general service time sequence, provided that the component arrival time and service time sequences satisfy FLDPs with rate functions satisfying the conditions of Theorems 3.1 and 3.2.

6. Large Deviations for Departure Times

In order to obtain LDPs for waiting times of queues within an acyclic network, we need to obtain an LDP for departure processes. The departure time of the n^{th} customer and the associated normalized process are

$$d_n = \sum_{i=1}^{n} u_{i-1} + w_n + v_n \quad \text{and} \quad D_n(t) = n^{-1} d_{\lfloor nt \rfloor}, \ t \geq 0 \ . \tag{6.1}$$

Theorem 6.1. *Assume that the conditions of Theorem 5.1 (a) hold with $I_{U,V}(u,v) = \infty$ if either u or v is not continuous. Assume that the U_n have paths unbounded above with probability 1. (a) Then $\{(W_n, D_n),\ n \geq 1\}$ obeys an FLDP in $D \times E^{\uparrow}$ for the same topology with rate function*

$$I_{W,D}(w,d) = \inf_{\substack{u,v: \\ w=R(v-u) \\ d=w+u}} \{I_{U,V}(u,v)\} \ . \tag{6.2}$$

Moreover, $\{D_n, n \geq 1\}$ obeys an FLDP in $E^{\uparrow}$ for the same topology with rate function

$$I_D(d) = \inf_{\substack{u,v: \\ d=u+R(v-u)}} \{I_{U,V}(u,v)\} \ . \tag{6.3}$$

(b) If, in addition, the conditions of Theorem 5.1 (b) hold, then

$$\begin{aligned} I_{W,D}(w,d) \ = \ & \int_0^\infty 1_{\{w(t)>0\}} \left[\lambda_U(\dot{d}(t) - \dot{w}(t)) + \lambda_V(\dot{d}(t))\right] dt \\ & + \int_0^\infty 1_{\{w(t)=0\}} \left[\lambda_U(\dot{d}(t)) + \inf_{0 \leq y \leq \dot{d}(t)} \lambda_V(y)\right] dt \end{aligned} \tag{6.4}$$

if w and d are absolutely continuous, w is nonnegative with $w(0) = d(0) = 0$ and $I_{W,D}(w,d) = \infty$ otherwise.
(c) If, in addition, the conditions of Theorem 5.1 (c) hold, then

$$\begin{aligned} I_{W,D}(w,d) \ = \ & \int_0^\infty 1_{\{w(t)>0\}} \sup_{\alpha,\beta} \{\beta \dot{d}(t) + \alpha(\dot{d}(t) - \dot{w}(t)) - \psi_U(\alpha) - \psi_V(\beta)\} dt \\ & + \int_0^\infty 1_{\{w(t)=0\}} \sup_{\alpha,\beta} \{(\alpha + \beta \wedge 0)\dot{d}(t) - \psi_U(\alpha) - \psi_V(\beta)\} dt \ . \end{aligned} \tag{6.5}$$

In general, an explicit expression for the rate function $I_D(d)$ seems difficult to obtain analytically from (6.2)–(6.5). However, they provide a basis for calculating the rate function $I_D(d)$ numerically. We also can deduce the form of the rate function in special cases. In particular, we now consider deterministic service times, as in [3], [17]. The rate function of the departure times then is identical to the rate function of the arrival times, in the region where it is finite. The following result enables us to obtain FLDPs for waiting times at each queue of an acyclic network of queues with all service times deterministic.

Corollary 1. *If, in addition to the assumptions of Theorem 6.1 (b), the service times are deterministic, so that $\lambda_V(x) = 0$ for $x = v$ and $\lambda_V(x) = \infty$ otherwise, and $\lambda_U(u) = 0$ for $u \geq v$ $(\rho \leq 1)$, then*

$$I_D(d) = \int_0^\infty \lambda_D(\dot{d}(t)) dt \ , \tag{6.6}$$

where $\lambda_D(z) = \lambda_U(z)$ *if* $z \geq v$ *and* $\lambda_D(z) = \infty$ *if* $z < v$.

We now consider the case of deterministic interarrival times. This case illustrates that the infimum of (6.4) over w need not be attained at $w(t) = 0$, $t \geq 0$. This case also demonstrates that the rate function I_D cannot always be written as the integral of a local rate function λ_D that depends only on the derivative $\dot{d}(t)$. In this case we can use a local rate function that is a function of the two variables $\dot{d}(t)$ and $d(t) - ut$.

Corollary 2. *In addition to the assumptions of Theorem 6.1 (b), assume that the interarrival times are deterministic, so that* $\lambda_U(x) = 0$ *for* $x = u$ *and* $\lambda_U(x) = \infty$ *otherwise, and that* $\lambda_V(v) = 0$ *for* $v \leq u$ *(*$\rho \leq 1$*). Then* $I_D(d) = 0$ *when* $d(t) = ut$, $t \geq 0$, *and, provided that* $d(t) - ut \geq 0$ *for all* t,

$$I_D(d) = \int_0^\infty \lambda_D(d(t) - ut, \dot{d}(t))dt \ , \tag{6.7}$$

where

$$\lambda_D(y, z) = \lambda_V(z)1_{\{y>0\}} + \inf_{0 \leq x \leq z} \lambda_V(x)1_{\{y=0\}} \ . \tag{6.8}$$

Otherwise $I_D(d) = \infty$.

In Corollary 2, in order to obtain a finite rate function the derivative $\dot{d}(t)$ is required to alternate between intervals on which it is equal to u (which contribute 0 to $I_D(d)$) and intervals over which its averages starting from the left end point of the interval exceed u. The number of intervals on which $\lambda_D(\dot{d}(t))$ can be positive can be finite or infinite. To illustrate, if $\dot{d}(t) = 2u1_{[0,1)}(t) + u1_{[2,\infty)}(t)$, then $I_D(d) = \lambda_V(2u) + \lambda_V(0)$. On the other hand, if $\dot{d}(t) = 2u1_{[1,2)}(t) + u1_{[2,\infty)}(t)$, then $I_D(d) = \infty$.

Given FLDPs for the departure times in Corollaries 1 and 2 to Theorem 6.1, we obtain FLDPs for the corresponding continuous-time departure processes by applying Section 3.

We tend to have more arrivals when the interarrival times are small. The following result gives a general result for small interarrival times.

Corollary 3. *In the general setting of Theorem 6.1 (b), if* $\dot{d}(t) = c < u$ *for* $0 \leq t < t_0$ *and* $d(t) = u$ *for* $t \geq t_0$, *then*

$$I_D(d) = t_0\lambda_D(c) \tag{6.9}$$

for

$$\lambda_D(z) = \lambda_U(z) + \inf_{0 \leq y \leq z} \lambda_V(y) \ . \tag{6.10}$$

References

[1] Anantharam, V. (1989). How Large Delays Build Up in a GI/G/1 Queue. *Queueing Systems* **5**, 345–367.

[2] Billingsley, P. (1968). *Convergence of Probability Measures.* Wiley, New York.

[3] Chang, C. S. (1995). Sample Path Large Deviations and Intree Networks. *Queueing Systems* **20**, 7–36.

[4] Chang, C. S. and Thomas, J. A. (1995). Effective Bandwidths in High-Speed Digital Networks. *IEEE J. Sel. Areas Commun.* **13**, 1091–1100.

[5] Duffield, N. G. and O'Connell, N. O. (1995). Large Deviations and Overflow Probabilities for the General Single Server Queue, with Applications. *Math. Proc. Camb. Phil. Soc.* 118, 363–374.

[6] Duffield, N. G. and Whitt, W. (1996). Large Deviations for Inversely Related Processes with Nonlinear Scaling, in preparation.

[7] Glynn, P. W. and Whitt, W. (1994). Logarithmic Asymptotics for Steady-State Tail Probabilities in a Single-Server Queue. *J. Appl. Prob.* **31A**, 131–156.

[8] Glynn, P. W. and Whitt, W. (1994). Large Deviations Behavior of Counting Processes and their Inverses. *Queueing Systems* **17**, 107–128.

[9] Iglehart, D. L. and Whitt, W. (1970). Multiple Channel Queues in Heavy Traffic, I and II. *Adv. Appl. Prob.* **2**, 150–177 and 355–369.

[10] Lynch, J. and Sethuraman, J. (1987). Large Deviations for Processes with Independent Increments, *Ann. Prob.*, **15**, 610–627.

[11] Puhalskii, A. (1991). On Functional Principle of Large Deviations. *New Trends in Probability and Statistics*, V. Sazonov and T. Shervashidze (eds.), VSP/Mokslas **1**, 198–218.

[12] Puhalskii, A. (1994). Large Deviations of Semimartingales Via Convergence of the Predictable Characteristics. *Stochastics* **49**, 27–85.

[13] Puhalskii, A. (1995). Large Deviation Analysis of the Single Server Queue. *Queueing Systems* **21**, 5–66.

[14] Puhalskii, A. and Whitt, W. (1996). Functional Large Deviation Principles for First Passage Time Processes, submitted

[15] Puhalskii, A. and Whitt, W. (1996). Large Deviation Principles for Waiting Times and Departure Times in Single-Server Queues, in preparation.

[16] Skorohod, A. V. (1956). Limit Theorems for Stochastic Processes. *Theor. Probability Appl.* **1**, 261–290.

[17] de Veciana, G., Courcoubetis, C. and Walrand, J. (1994). Decoupling Bandwidths for Networks: A Decomposition Approach to Resource Management for Networks. *Proc. IEEE Infocom '94*, **2**, 466–474.

[18] de Veciana, G., Kesidis, G. and Walrand, J. (1995). Resource Management in Wide-Area ATM Networks Using Effective Bandwidths. *IEEE J. Sel. Areas Commun.* **13**, 1081–1090.

[19] Whitt, W. (1980). Some Useful Functions for Functional Limit Theorems. *Math. Opns. Res.* **5**, 67–85.

Sharing Bandwidth in ATM

Kavita Ramanan
Brown University

Alan Weiss
Bell Laboratories—Lucent Technologies

Abstract

We present and asymptotically analyze a model of a communication channel, where sources are on-off, and share as much of the available bandwidth as possible. The asymptotic regime is a large number of sources and a proportionally fast channel. Our main results are a fluid limit theorem and a large deviations limit theorem. We also discuss some practical consequences of our analysis.

Introduction

What are the benefits of allowing users to share all the bandwidth in a communication channel? Presumably the channel is more fully utilized than when users get only a fixed bandwidth. Users should get more throughput and experience less delay, and the channel should be able to support more users at a given quality of service. In this paper we quantify these benefits in a simple model of sharing. We model a large channel serving many statistically identical users. Our model is intended to represent ABR (Available Bit Rate) service, which is a proposed service on ATM networks.

Our analysis follows the method and notation of Chapter 13 of Shwartz and Weiss [SW]. The analysis is asymptotic in n, a parameter that represents the number of users, and also scales both the capacity of the channel and the total amount of buffering (if any). This scaling was also used in Botvich and Duffield et al. [BD] to scale ATM models, though the analysis was different. Courcoubetis and Weber [CW] also analyze ATM systems with this scaling by a different method. In contrast, most other analyses of ATM asymptotics treat the buffer as large, with the number of users held fixed [Kesidis et al. [KWC], Gibbens and Hunt [GH], deVeciana et al. [VOW], Tse et al. [TGT], Guibert and Simonian [GS], Elwalid and Mitra [EM]]. Altman et al. [ABB] considered the benefits of sharing in ATM. Furthermore, Queija and Boxma [QB] analyzed the benefits in the context of sharing bandwidth amongst ABR and VBR sources.

Our model applies to a particular time scale. We model a group of sources sharing bandwidth as if they divide all the bandwidth evenly, and instantaneously adjust to changing conditions. In reality, there would exist some feedback mechanism to allow the sources to change their rates. Since this happens instantaneously in our model, our model only applies to time scales much longer than a feedback mechanism would take (typically a round-trip delay, on the order of 50 msec. across the continental US). Also, we assume that an active connection might experience many adjustments in bandwidth during its lifetime. That is, we are modeling file transfers that typically take many multiples of a feedback time.

Our analysis has the usual strengths and weaknesses of large deviations sample-path asymptotics. Since it gives explicit estimates for the quantities of interest, there are no computational issues, irrespective of the size of the parameter n. The explicit nature of the

formulae also allows us to analyze the effects of modifying any of the parameters in the model. However, there are no useful bounds on the errors of the estimates. We did not perform any simulations to estimate these errors. We obtain explicit sample path information on the manner of occurrence of rare events, though again, without useful error bounds.

As a final point before beginning our analysis, we wish to point out an advantage and a weakness to our approach that are perhaps not sufficiently well known. Our analysis is quite similar to that in [W] and [SW]. Yet the current model is different: it is not time-reversible, it does not have a representation as a simple reflection map, and it has hysteresis (these features are all manifestations of one aspect of the model). The ease with which we can treat the model shows the *robustness* of sample path large deviations combined with a Freidlin-Wentzell approach to steady state, especially as contrasted with the difficulty of extending other approaches. However, our reliance on Freidlin-Wentzell theory makes it unlikely that our approach could be modified to be more accurate. This conference paper contains no details. We plan to report the proofs in [RW].

Model and Notation

Our model consists of n traffic sources, a buffer to which they feed fluid, and a server. The server is the simplest part of the system; it empties the buffer at a constant rate nC whenever the buffer content $b(t) > 0$. When $b(t) = 0$, the server empties incoming fluid immediately as long as the incoming rate is no more than nC. If the incoming rate exceeds nC, then the buffer fills at the rate (incoming - nC). The buffer may be finite (size nB) or infinite—our analysis is valid for both cases.

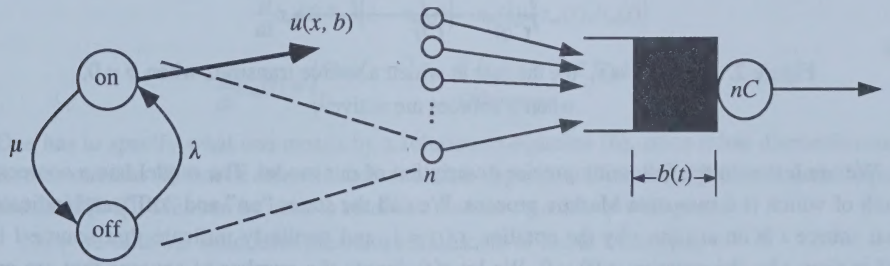


Figure 1. A basic model of ATM sources with finite buffer.

The behavior of the sources is more complicated, so we begin with an intuitive explanation. There are two rates $r \leq R$ that respectively represent the minimum and maximum bandwidths that an active source gets. For our purpose, the bandwidth assigned to a source is the fluid rate the source delivers to the buffer.. Sources alternate between active and quiescent states. The idea is that at the end of each quiescent state, the source generates a message to be sent. The time taken to send the message depends on its length and the bandwidth the source uses. After the message is sent, the source again becomes quiescent. We assume, purely for mathematical convenience, that the lengths of messages are i.i.d. exponentially distributed random variables with mean length one. The time it takes a source to transmit a

message depends on the bandwidth available. If there are few other active sources, the message can be transmitted at rate R, and so takes an exponentially distributed amount of time to transmit, with mean duration $1/R$. If the buffer is not empty (so the server is serving at full rate), or if many sources are active, then the message is transmitted at rate r, thus taking an average duration of $1/r$ to transmit. If the buffer is empty and there are a moderate number x of sources active, then each source shares the bandwidth equally, hence obtaining a rate nC/x.

There is one further complication in our model. The buffer has priority over new traffic when it is not empty. That is, when the buffer contains fluid, all active sources receive r bandwidth, and the buffer changes at rate $xr - C$. If $xr - C < 0$ then the buffer drains, while if $xr - C > 0$, then the buffer fills. Each active source will receive a rate r until the buffer is empty. This complication causes some mathematical inconvenience, since processes with hysteresis have not been fully covered by existing theory. Nonetheless, we feel that it is a reasonable and realistic feature of our model.

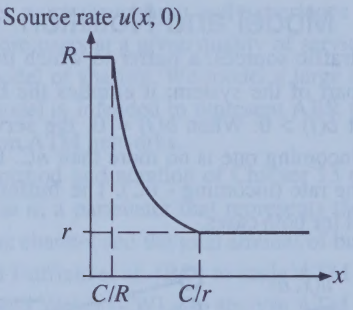


Figure 2. A plot of $u(x, 0)$, the rate at which a source transmits when $b = 0$, when x sources are active.

We are left with the following precise description of our model. The model has n sources, each of which is a two-state Markov process. We call the states "on" and "off", and indicate that source i is on at time t by the notation $x_i(t) = 1$, and similarly indicate that source i is off at time t by the notation $x_i(t) = 0$. We let $x(t)$ denote the number of sources that are on at time t; that is,

$$x(t) \equiv \sum_{i=1}^{n} x_i(t). \tag{1}$$

We let $b(t)$ denote the content of the buffer at time t. For the moment, we suppose that the buffer is potentially infinite. We define an auxiliary function v:

$$v(x,b) \equiv \begin{cases} rx - nC & \text{if } b > 0 \text{ or } rx > nC \\ 0 & \text{Otherwise} \end{cases} \tag{2}$$

Now defining the bandwidth $u(x, b)$ by the equation

$$u(x,b) = \begin{cases} r & \text{if } b > 0 \text{ or } rx > C \\ R & \text{if } b = 0 \text{ and } Rx < C \\ \frac{C}{x} & \text{if } b = 0 \text{ and } \frac{C}{R} \le x \le \frac{C}{r} \end{cases} \quad (3)$$

we have the following equation for the generator L of the process $(x(t), b(t))$:

$$Lf(x,b) = \lambda(n-x)(f(x+1,b) - f(x,b)) + xu(x/n,b)(f(x-1,b) - f(x,b)) + v(x,b)\frac{\partial}{\partial b}f(x,b) \quad (4)$$

That is, the buffer satisfies the differential equation $\frac{d}{dt}b(t) = v(x(t),b(t))$, and $x(t)$ is a birth-death process with birth rate $\lambda(n - x)$ and death rate $xu(x/n, b)$.

Fluid Analysis

By far the easiest and most useful part of our analysis is the fluid analysis. This is a law of large numbers scaling applied to a stochastic process; in other words, it is a deterministic differential equation, whose solutions approximate the sample paths of a stochastic process. We scale time and space as follows:

$$\begin{gathered} z_n(t) = x(t)/n \\ b_n(t) = b(t)/n \\ \text{number of sources} = n \\ \text{Service rate} = nC \end{gathered} \quad (5)$$

All other parameters of the process remain unchanged. Then $(z_n(t), b_n(t))$ is a Markov process, which we show to be nearly deterministic. We first define the fluid limit of $(z_n(t), b_n(t))$ to be the solution of:

$$\begin{gathered} \frac{d}{dt}z_\infty(t) \equiv \lambda\left(1 - z_\infty(t)\right) - z_\infty(t)u\left(z_\infty(t), b_\infty(t)\right) \\ \frac{d}{dt}b_\infty(t) \equiv \begin{cases} rz_\infty(t) - C & \text{if } b_\infty(t) > 0 \text{ or } rz_\infty(t) > C \\ 0 & \text{Otherwise} \end{cases} \end{gathered} \quad (6)$$

One has to specify what one means by a solution to equation (6), since it has discontinuities. It turns out that (6) can be reformulated as a state-dependent Skorokhod problem (see Dupuis and Ishii [DI]). This can be used to establish that the equation is well-posed.

Now we state our fluid limit theorem, reminiscent of Kurtz's theorem:

Theorem 1: *For any $T > 0$ and $\varepsilon > 0$, there are positive constants K and J such that for any $x \in [0, 1]$, $b \ge 0$, and $n > 1$,*

$$P_{x,b}\left(\sup_{0 \le t \le T}\left|z_n(t) - z_\infty(t)\right| + \sup_{0 \le t \le T}\left|b_n(t) - b_\infty(t)\right| > \varepsilon\right) \le Ke^{-nJ}$$

This theorem states that uniformly over starting positions of the buffer and process and scale factor n, the scaled buffer and process are extremely likely to be close to their fluid approximations, which are defined by equation (6). The subscript (x, b) refers to the starting position of the process $(z_n(t), b_n(t))$ and of its fluid limit $\left(z_\infty(t), b_\infty(t)\right)$. Note that not every x is a possible starting point for every n; it turns out that the theorem holds even if we let the starting point of the process be the nearest multiple of $1/n$ to x.

Theorem 1 has many uses. First, it is very easy to analyze $\left(z_\infty(t), b_\infty(t)\right)$, since it is easy to find a solution to equation (6). Therefore, we can obtain information about the process of

interest $(z_n(t), b_n(t))$ by examining how $(z_\infty(t), b_\infty(t))$ behaves. Secondly, we can get an idea of how $(z_n(t), b_n(t))$ behaves in steady state, even though Theorem 1 is ostensibly only about finite time behavior. This is because, under a simple condition, all solutions of equation (6) approach the same limit point exponentially quickly in time. This is enough to give strong results on the steady-state distribution of $(z_n(t), b_n(t))$.

Estimates of Transient Behavior

The transient behavior of $(z_\infty(t), b_\infty(t))$ has two different characters, depending on the following condition.

Stability Condition: $$\frac{\lambda r}{\lambda + r} < C \tag{7}$$

Equation (6), the defining equation for $(z_\infty(t), b_\infty(t))$, is nearly a linear differential equation. The only irregularities come from the discontinuities in the functions v and u. The following lemma makes the whole calculation transparent.

Lemma 1: *Suppose that the stability condition (7) is satisfied. Then for any starting state, $b_\infty(t)$ has exactly one period of time where it is nonzero, and this period occurs immediately after time 0.*

Lemma 1 makes it easy to calculate $(z_\infty(t), b_\infty(t))$, because we only have to check at the beginning to see if $b_\infty(t)$ is zero or not. If it is nonzero, then $u = r$, and equation (6) is linear until such time as $b_\infty(t)$ becomes zero. Thereafter, $u = 0$, and we can simply calculate (6) for $z_\infty(t)$. This leads to the following theorem:

Theorem 2: *Suppose that the stability condition (7) is satisfied. Then for any starting state, $(z_\infty(t), b_\infty(t))$ approaches the point $(z^*, 0)$ exponentially quickly in time, where*

$$z^* = \begin{cases} \dfrac{\lambda}{\lambda + R} & \text{if } \dfrac{\lambda R}{\lambda + R} < C \\ \dfrac{\lambda - C}{\lambda} & \text{if } C \le \dfrac{\lambda R}{\lambda + R} \end{cases} \tag{8}$$

In fact, $b_\infty(t)$ becomes zero after a finite time.

When the stability condition is violated, the system is overloaded, in the sense of Theorem 2a.

Theorem 2a: *Suppose that $\dfrac{\lambda r}{\lambda + r} > C$. Then for any starting state, $b_\infty(t)$ approaches ∞.*

Steady-State Estimates

When the stability condition is satisfied, we see that there is a unique point z^* such that $z_n(t)$ tends to z^* for sufficiently large t. We now state a precise theorem about what this means for steady state. It may seem surprising that we do not need a full large deviations principle and Freidlin-Wentzell theory to bootstrap a finite time result to a steady-state one. The argument we use follows one in [SW] that shows that when a process lives on a compact state space, Kurtz's theorem already implies something about the steady-state distribution. While our process $z_n(t)$ is not on a compact state space since it depends on $b_n(t)$, which has

no upper bound in general, it is close enough to bounded for the argument to go through.

Theorem 3: *If the stability condition (7) holds, then for large enough n the process* $(z_n(t), b_n(t))$ *has a proper steady state distribution. As n becomes large, this distribution converges weakly to a delta-distribution at the point* $(z^*, 0)$.

The idea behind the proof of Theorem 3 is similar to a Freidlin-Wentzell argument. Choose a large T and small ε, and let n be large enough that the probability given in Theorem 1 is very small. Then over each period of time of length T, $z_n(t)$ is very likely to stay quite close to z^*, and so $b_n(t)$ gets close to zero if it started away from zero, and stays equal to zero if it did start at zero. All errors are exponentially small in n.

Theorem 3 is the basis for most of the practical implications we can draw about our model of ABR traffic. It states that the steady-state fraction of sources that are active is deterministic (or nearly so), and thus each source gets a deterministic share of the bandwidth. Here are some of the engineering consequences, assuming that the stability condition holds.

1. The channel is fully utilized if $C \le \frac{\lambda R}{\lambda + R}$; that is, the limiting factor in the speed of communication is the speed of the channel nC.

2. The channel is partially utilized if $\frac{\lambda R}{\lambda + R} < C$; that is, the limiting factor in the speed of communication is the access speed R.

3. The time it takes a source to send one message is approximately exponentially distributed with parameter $u(z^*, 0)$.

Large Deviations

It should not be surprising to find that the process $(z_n(t), b_n(t))$ satisfies a large deviations principle. In this section we write the rate function, state the large deviations principle, and give a hint about the technical difficulties involved in proving the principle. Then, in a show of utilitarianism, we solve both the upcrossing problem and the buffer-crossing problem, and show the engineering consequences of our solution.

Large Deviations Theory

Our process $(z_n(t), b_n(t))$ is similar to the process of the same name in Chapter 13 of [SW]. The main differences as far as proving theorems are the presence of hysteresis, and the more discontinuous nature of the equation for $b_n(t)$. The former requires some careful examination of the proof of the upper bound. The latter requires one to be careful in proving the lower bound. We will say just a bit more about these points after we state the rate function and the upper and lower bounds.

We define the local rate function as follows:

$$h(z,b,y,a) = \begin{cases} f\big(y,(1-z)\lambda,zu(z,b)\big) & \text{if } a = v(z,b),\ 0 \le z \le 1,\ 0 \le b \\ \infty & \text{otherwise} \end{cases} \tag{9}$$

where

$$f(y,\lambda,\mu) = y \log\left(\frac{y + \sqrt{y^2 + 4\lambda\mu}}{2\lambda}\right) + \lambda + \mu - \sqrt{y^2 + 4\lambda\mu} \tag{10}$$

and recall $v(z, b)$ was defined in equation (2). The rate function for the process $(z_n(t), b_n(t))$ is now defined on absolutely continuous pairs of real-valued functions $(z, b)(t)$ by

$$I_0^T(z,b) = \int_0^T h\left(z, b, \frac{d}{dt}z, \frac{d}{dt}b\right)dt \tag{11}$$

Intuitively, this rate function is the usual one for a birth-death process z with birth rate $\lambda(1 - z)$ and death rate $zu(z, b)$, and where $b(t)$ satisfies an ordinary differential equation with v as its derivative. We know that, locally, this is exactly how the process behaves. The only difficulty lies in proving that this guess is correct, since there are discontinuities and hysteresis.

We now state the upper and lower bounds that form the large deviations principle for our process $(z_n(t), b_n(t))$. The state space is $D[0,T]\times R^+[0,T]$ for some fixed T, where $D[0, T]$ is the standard Skorokhod space, and $R^+[0,T]$ is the space of nonnegative real functions with the sup norm topology. We represent nonnegative numbers by R^+.

Theorem 4. *Given a compact set C in $[0,1]\times R^+$, for each closed set F in $D[0,T]\times R^+[0,T]$,*

$$\limsup_{n\to\infty} \frac{1}{n}\log P_x\left((z_n,b_n)\in F\right) \le - \inf_{(z,b)\in F} I_0^T(z,b) \tag{12}$$

uniformly over starting points in C.

Theorem 5. *Given a compact set C in $[0,1]\times R^+$, for each open set G in $D[0,T]\times R^+[0,T]$,*

$$\liminf_{n\to\infty} \frac{1}{n}\log P_x\left((z_n,b_n)\in G\right) \ge - \inf_{(z,b)\in G} I_0^T(z,b) \tag{13}$$

uniformly over starting points in C.

The upper bound is proved almost exactly as in [DEW]. In fact, the main difficulty is in identifying the rate function, and seeing that the theorem applies even in the event of hysteresis. The lower bound is proved in the usual way via change of measure as in [SW, chapter 8]. The only subtlety lies in the discontinuities of the underlying differential equations, but since the Skorokhod problem turns out to be well posed, this difficulty vanishes.

We state, without even a precise theorem, that the Freidlin-Wentzell theory applies to our process $(z_n(t), b_n(t))$. That is, we can estimate not only the most likely point for the invariant measure, but we can estimate its tails, too, by the rate function.

Large Deviations Applications

In this section we solve some variational problems associated with our process, and compare the problems and solutions to those of the well-studied AMS model. To begin, we must describe the AMS model, and what is known about it in this context.

The briefest description of the AMS model is that it is the present model with $r = R$. That is, when sources are on, they transmit at rate r, regardless of the state of the buffer or the other sources. There is no hysteresis, and there are fewer discontinuities in the associated differential equations. Chapter 13 of [SW] contains an analysis of the AMS model that we use as the base for this paper.

If we let $r = 1, \mu = r$, and define

$$q = \frac{\lambda}{\lambda + \mu} \tag{14}$$

(q represents the point z^* for the AMS model), then the rate at which the buffer becomes nonempty is

$$I^* = \int_q^C \log \frac{\mu x}{\lambda(1-x)} dx = C \log \frac{C}{q} + (1-C) \log \frac{1-C}{1-q} \tag{15}$$

In other words, the probability that the buffer is nonempty is about e^{-nI^*}. Furthermore, there is a function $I(B)$ such that the probability in steady state that the buffer exceeds a level B is about $e^{-n(I^* + I(B))}$. The function $I(B)$ is given in equation (13.41) of [SW] (there it incorporates I^* also in the form of a constant c_1), and is too complicated to reproduce here. The specific statement that we claim holds for $I(B)$ is

$$\lim_{n \to \infty} \frac{1}{n} \log P_{ss}\{b_n(t) \geq B\} = -I^* - I(B) \tag{16}$$

where P_{ss} refers to steady-state probability. We note that $I(B)$ has simple asymptotic forms for B large or small.

Now our problem at hand is to estimate the behavior of the buffer in the current model. The following theorem tells us all we need to know, from a large deviations point of view.

Theorem 6. *The probability that the buffer $b_n(t)$ exceeds a level B in steady state is given by the rate function $I_1^* + I(B)$ in the sense of equation (16). That is, asymptotically, the current model and the AMS model have the same buffer statistics to within a constant factor.*

The constant I_1^* referred to in Theorem 6 is given by the following equation:

$$I_1^* = \int_{z^*}^{\frac{C}{r}} \log \frac{xu(x,0)}{\lambda(1-x)} dx \tag{17}$$

Comparing (17) and (15), and noting that $u(x, 0) > r$ in the region of integration, and that $z^* < q$, we see that

$$I_1^* > I^* \tag{18}$$

We can explicitly calculate I_1^*. It simply depends on whether or not $z^* > C/R$. By equation (8), we have the following:

$$I_1^* = \begin{cases} \left(1 - \frac{C}{r}\right) \log\left(1 - \frac{C}{r}\right) + \frac{C}{r} \log \frac{C}{\lambda} - \log \frac{R}{\lambda + R} + \frac{C}{r} - \frac{C}{R} & \text{if } z^* < \frac{C}{R} \\ \left(1 - \frac{C}{r}\right) \log\left(1 - \frac{C}{r}\right) + \frac{C}{r} \log \frac{C}{\lambda} - \log \frac{C}{\lambda} - 1 + \frac{C}{\lambda} + \frac{C}{r} & \text{if } z^* \geq \frac{C}{R} \end{cases} \tag{19}$$

In short, we have the following result for buffer asymptotics. Letting P_{AMS} represent steady-state probability in the AMS model, and P_{ss} represent steady-state probability in the current model, we claim that

$$\lim_{n \to \infty} \left[\frac{1}{n} \log \frac{P_{AMS}\{b_n(t) \geq B\}}{P_{ss}\{b_n(t) \geq B\}} \right] = I_1^* - I^* \tag{20}$$

Engineers often set overflow probabilities of the order of 10^{-9} for ATM traffic. Our result

can be used to give a guide for determining the capacity and mean throughput of a system that meets such a performance objective. For example, an engineer could specify a service rate nC, and service parameters r and R. Then one finds the approximate overflow probability using equations (15) through (20). One then adjusts n and C, keeping nC constant, so that the performance objective is met. This is straightforward since I^* is monotone in C. The resulting mean throughput is then given by the fluid approximation, as explained in the final three points of that section.

References

[ABB] Altman, E., F. Baccelli, J.-C. Bolot, *Discrete-Time Analysis of Rate-Based Feedback Control Mechanisms*, Preprint, 5th Int'l. Conf. on High Speed Networks, Raleigh, Oct. 1993.

[BD] Botvich, D.D. and N.G. Duffield, *Large deviations, the shape of the loss curve, and economies of scale in large multiplexers*, Queueing Systems, **20**, pp. 293-320, 1995.

[CW] Courcoubetis, C. and R. Weber, *Effective Bandwidths for Stationary Sources*, Probability in Engineering and Information Sciences, 1996.

[DEW] Dupuis, P., R.S. Ellis, A. Weiss, *Large deviations for Markov processes with discontinuous statistics I: General upper bounds*, Ann. Prob. **19**, pp. 1280–1297, 1991.

[DI] Dupuis, P. and H. Ishii, *On Lipschitz continuity of the solution mapping to the Skorokhod problem, with applications*, Stochastics and Stoch. Reports, **35**, pp. 31–62, 1990.

[EM] Elwalid, A.I., and D. Mitra, *Effective Bandwidth of General Markovian Traffic Sources and Admission Control of High Speed Networks*, IEEE/ACM Trans. Networking, **1**, pp. 329–343, 1993.

[GH] Gibbens, R.J., and P.J. Hunt, *Effective bandwidths for the multi-type UAS channel*, QUESTA, **9**, pp. 17–28, 1991.

[GS] Guibert, J. and A. Simonian, *Large deviations approximation for fluid queues fed by a large number of on/off sources*, IEEE JSAC, **13**, pp. 1017–1027, 1995.

[KWC] Kesidis, G., J. Walrand, and C.S. Chang, *Effective Bandwidths for multiclass Markov Fluids and other ATM Sources*, IEEE/ACM Trans. Networking, **1**, pp. 424–428, 1993.

[QB] Queija, R.N. and O.J. Boxma, *Analysis of a multi-server queueing model of ABR*, Preprint, 1996.

[RW] Ramanan, K. and A. Weiss, *A model of ABR traffic via fluid limits and large deviations*, manuscript in preparation.

[SW] Shwartz, A. and A. Weiss, *Large Deviations for Performance Analysis*, Chapman and Hall, London, 1995.

[TGT] Tse, D.N.C., R. G. Gallager, and J.N. Tsitsiklis, *Statistical Multiplexing of Multiple Time-Scale Markov Streams*, IEEE JSAC, **13**, 1995.

[VOW] de Veciana, G., C. Olivier, J. Walrand, *Large Deviations of Birth Death Markov Fluids,* Probability in the Engineering and Informational Sciences, **7**, pp. 237–255, 1993.

[W] Weiss, A., *A new technique for analyzing large traffic systems*, Adv. Appl. Prob. **18**, pp. 506–532, 1986.

On the relevance of long-tailed durations for the statistical multiplexing of large aggregations.

N. G. Duffield
AT&T Laboratories
Room 2C-323, 600 Mountain Avenue, Murray Hill, NJ 07974, USA
duffield@research.att.com

Abstract

In this paper we obtain large deviation bounds for the queue length in a buffer fed by an $M/G/\infty$ process with G long-tailed, and in the limit that the arrival intensity grow to infinity along with the service rate so as to keep the offered load constant. This provides an example in which long-tailed arrival distributions are no bar to obtaining statistical multiplexing gain when such arrivals are sufficiently spread out over time.

1 Introduction.

In this paper we investigate the extent to which long tailed behavior in the activity of sources make them difficult to statistically multiplex together. The word *together* is crucial here, because it is easier to achieve multiplexing gain across several sources than within a single source.

To achieve gain within a source at a buffer requires being able to allocate to it a service rate (between its peak and mean rate) in such a way that statistical fluctuations in the arrivals overflow the buffer sufficiently rarely. Much recent work has been devoted to finding large deviation approximations to the probability that, in an infinite buffer, the queue length Q exceeds a given level b. Such approximations can be written informally as

$$\mathbf{P}[Q > b] \approx e^{-h(b)\delta}, \tag{1}$$

where the scaling function h is determined by the arrival process. For a large class of short range dependent processes (see [11, 10] and references in [4]) h is linear, and so the tail is exponential. For class of long-range dependent arrival processes, including fractional Brownian motion, h is a power law and so the tail is Weibullian (see [8] and a previous lower bound in [15]).

The $M/G/\infty$ process has recently attracted some attention as a model of arrivals (see e.g. [1, 13, 18]). It is a random marked Poisson process of some intensity r the marks of which are i.i.d. random variables with common distribution G with mean σ. The arrival times are those of customers, each of which commences service immediately for a time whose duration is the corresponding mark. Let N_t be number of customers in the system at time t and define $A_t = \int_0^t dt' N_{t'}$. The interpretation of A as an arrivals process is that each of the N_t customers present at time t deposits fluid into the buffer at unit rate (or

in discrete time, packets of unit size at each timeslot). The fluid is drained out at rate s. Set $\phi = s - r\sigma > 0$ for stability.

Parulekar and Makowski [16] have determined an asymptotic of the form (1) for this model when G is short tailed. Building on their work, we find a large deviation *upper bound* when G is Pareto with power $\alpha \in (1,2)$:

$$\limsup_{b\to\infty} \frac{\log \mathbf{P}[Q > b]}{\log b} \leq 1 - (\alpha - 1)\phi, \tag{2}$$

when the RHS is negative. We have not determined whether this bound is tight: as we discuss in section 4 below, the large deviation lower bound does not obtain in this case, at least not by use of the methods in [8]. Observe though that the exponent in (2) is different to that which would be found in the $M/G/1$ model or On-Off Model (see [5]) with the same service distribution, namely an exponent of $1 - \alpha$. That there is difference is not surprising, since large deviations in the two models occur by different mechanisms. For the $M/G/1$ model, since G is long tailed, the most likely way for an (asymptotically large) backlog of work to arise is for one of the service periods to be large. However, in the $M/G/\infty$ model, to develop a backlog of size b requires that a number $n > s$ of sources to be active during a period of duration at least $b/(n-s)$: some "conspiracy" between arrivals is required. (This observation can be used to formulate lower bounds for $\mathbf{P}[Q > b]$).

This leads as to consider a second asymptotic: that of scaling the service rate as sL and the arrival rate as rL (or equivalently, superposing L sources of arrival rate r) with $L \to \infty$. In this limit the load remains independent of L, but the proportional contribution of each active source to that load goes to zero as $L \to \infty$. Scaling a buffer level likewise as Lb so as to keep the buffer size per unit arrival rate constant, we find that the requirement for conspiracy as the means of generating large deviations in the corresponding queue length Q^L becomes manifest: applying a result from [7] we show that

$$\lim_{L\to\infty} \frac{1}{L} \log \mathbf{P}[Q^L > Lb] = -I(b), \quad \text{with} \quad \lim_{b\to\infty} \frac{I(b)}{\log b} = (\alpha - 1)\phi. \tag{3}$$

Such asymptotics arise because for each fixed time t, the distribution of the normalized arrival process $L^{-1}A_t^L$ converges exponentially fast to its mean $r\sigma t$ as $L \to \infty$. From (3) we have the approximation

$$\mathbf{P}[Q^L > Lb] \approx e^{-LI(b)} \approx b^{-L(\alpha-1)\phi} \tag{4}$$

for first L and then b sufficiently large. For fixed b the tail probabilities are exponentially small in L, reflecting the multiplexing gain available across sources, while it decreases only algebraically in b: the effect of the long tailed service distribution. It is worth remarking that superpositions of Markovian sources or fractional Brownian Motions have the same exponential decay in L for their tail distributions, though $I(b)$ is asymptotically proportional to δb or δb^{2-2H} respectively (here H is the Hurst parameter of the fractional Brownian Motion).

That the long tailed on duration of the sources should not prevent statistical multiplexing gain even for long tailed active periods can be seen by observation similar to those made for the large b asymptotic. A small number of sources may effectively remove some of the service capacity during their long active periods, but it requires a large number of them to be active in order for a large deviation in the buffer occupancy proportional to the aggregate arrival rate. Such a heuristic does not evidently apply to the corresponding

$M/G/1$ model: in this case work arrives instantaneously in long tailed chunks, rather than being spread out over the duration of the active period: just one such arrival can use up the available service rate. Of course, from the point of view of modeling, one will chose $M/G/1$ or $M/G/\infty$ as appropriate: there is no necessary relation between them. But although both can be seen as members of the class of "arrival processes with long-tailed behavior", they have radically different properties under statistical multiplexing. In fact, rather than the decay of (4), $\mathbf{P}[Q^L > Lb]$ decays only algebraically in L as $L \to \infty$ for $M/G/1$.

The contents of the paper are as follows. In section 2 we recapitulate the large deviation lower bounds for the large buffer asymptotics for a single arrival process. In section 3 we extend a result in [8] to derive an upper bound for the large buffer asymptotics which we can apply to the long tailed $M/G/\infty$ process in section 4. We conclude in section 5 by deriving the tail asymptotics for the $M/G/\infty$ model as the arrival rate is increased to infinity while keeping the service duration and offered load constant.

Finally, we mention that although we have dealt here with loss probabilities, the duration of loss periods is also important for understanding the impact of long tailed durations. This has been studied in a related model in [1]. One expects that long tailed active period will give rises to long duration of periods of overload. However, for small overshoots of a given threshold, in bufferless models at least, the duration is overload can be largely independent of the duration of the active period beyond its mean: see [9] for more details.

2 Large deviation lower bounds

Let us formulate in more detail a large deviation result the tail asymptotics of the queue length in an infinite buffer which is fed by a stationary arrivals process and served at some constant rate s. Index time by $T = \mathbf{R}_+$ or $\mathbf{Z}_+$ and let A_t be the work arriving in the queue in the interval $[-t, 0)$ and define the (backward) excess workload process W by $W_0 = 0$ and $W_t = A_t - st$. Then we have the pathwise relation for the queue length Q at time 0 (see [3])

$$Q = \sup_{t \geq 0} W_t. \tag{5}$$

Here we shall be concerned with obtaining bounds of the form

$$-\delta_+ \leq \liminf_{b \to \infty} \frac{\log \mathbf{P}[Q > b]}{h(b)} \leq \limsup_{b \to \infty} \frac{\log \mathbf{P}[Q > b]}{h(b)} \leq -\delta_- \tag{6}$$

for some constants $\delta_\pm$ depending on the processes, and where h is a *scaling function*, namely a positive function on T which increases to $+\infty$. If $\delta_- = \delta_+ = \delta$ then we can informally write the approximation (1). Common to the works cited in the introduction is the idea of characterizing the large deviation properties of the queue length in term of those of the workload process.

Recall [6] that a *rate function* on $\mathbf{R}_+$ is a lower semicontinuous function taking values in $[0, \infty]$, and a *good* rate function is one with compact level sets. Let $(X_t : t \geq 0)$ be a real stochastic process and v a scaling function. Then the pair $(X_t, v(t))$ is said to satisfy a *large deviation principle* with rate function I if for all Borel subsets B of $\mathbf{R}_+$

$$-\inf_{x \in B^0} I(x) \leq \liminf_{t \to \infty} \frac{1}{v(t)} \log \mathbf{P}[X_t \in B] \leq \limsup_{t \to \infty} \frac{1}{v(t)} \log \mathbf{P}[X_t \in B] \leq -\inf_{x \in \bar{B}} I(x), \tag{7}$$

where B^o is the interior and $\bar{B}$ the closure of B.

According to the Gärtner-Ellis Theorem (see e.g. [6]), to conclude that $(X_t, v(t))$ satisfies and LDP it is sufficient to demonstrate the existence of the cumulant generating function (CGF, also called the log moment generating function) $\lambda(\theta) = \lim_{t\to\infty}(1/v(t)) \log \mathbf{E}[\exp(\theta v(t) X_t)]$, which in addition satisfies a (sufficient) technical condition, that it be essentially smooth. Then $(X_t, v(t))$ satisfies (7) with rate function equal to the Legendre transform λ^* of λ i.e.

$$\lambda^*(x) = \sup_{x\in\mathbf{R}}\{\theta x - \lambda(\theta)\}. \tag{8}$$

In the present case, define the transient CGF

$$\lambda_t(\theta) = \frac{1}{v(t)} \log \mathbf{E}[\exp(\theta v(t) W_t / a(t))]. \tag{9}$$

Theorem 1 *Assume*

(i) The limit $\lambda(\theta) = \lambda_t(\theta)$ exists as an extended real number for all $\theta \in \mathbf{R}$ and is essentially smooth.

(ii) $\lambda(\theta) < 0$ for some $\theta > 0$.

(iii) There exists a scaling function h such that the limit

$$f(c) = \lim_{t\to\infty} \frac{h(t)}{v(a^{-1}(t/c))} \tag{10}$$

exists and is finite for all $c > 0$ where $a^{-1}(x) = \sup\{t \in T : a(s) \le x\}$.

Then the lower bound in (6) holds with

$$\delta_+ \equiv \inf_{c>0} \frac{\lambda^*(c^+)}{f(c)} \quad \textit{which is equal to} \quad \inf_{c>0} \frac{\lambda^*(c)}{f(c)}, \tag{11}$$

if f is continuous, or the effective domain of λ^* is unbounded to the right.

Remark: One case in which (10) obtains easily is when $v \circ a^{-1}$ is *regularly varying* (see [2] for more details). In this case taking $h = v \circ a^{-1}$ we get (10) with $f(c) = c^{\tilde{f}}$ where the power $\tilde{f}$ is the *index* of the regularly varying function $v \circ a^{-1}$. Indeed, if $v \circ a^{-1}$ *is* regularly varying, this is the only choice of h which gives a non-trivial f.

Proof of Theorem 1. [8] gives us the lower bound in (6) with δ_+ as in (11). To remove the necessity of taking right limits, observe that being a convex function, λ^* is continuous on the interior of its effective domain (see Theorem 10.1 of [19]). Hence, as noted in [16], if the effective domain of λ^* contains $\mathbf{R}_+$, the second formulation of δ_+ in (11) applies. If not, the effective domain contains $[0, \bar{c})$ for some $\bar{c} \ge 0$. Due to (ii), λ^* is non-decreasing on $\mathbf{R}_+$. Hence

$$\inf_{c>0} \lambda^*(c^+)/f(c) = \inf_{0<c<\bar{c}} \lambda^*(c)/f(c) \tag{12}$$

where $\bar{c}$ is the boundary of the effective domain of λ^*. The second form of δ_+ now follows since f is continuous, and λ^* is continuous from the left at $\bar{c}$, being non-decreasing on $\mathbf{R}_+$ and lower semicontinuous. ∎

3 A Further Large Deviation Upper Bound.

Turning now to large deviation upper bounds for $\mathbf{P}[Q > b]$ we first formulate a bound independently of statistical assumptions on the workload process W. We work here in discrete time: extensions to continuous time can be made using argument similar to those in [8].

Theorem 2 *Let Δ be some function $\mathbf{R}_+ \to \mathbf{R}_+$, a, v, h scaling functions and λ_t as in (9).*

(i) Assume for all t that $\lambda_t(\theta) > 0$ for $\theta < 0$.

$$\limsup_{b\to\infty} \frac{\log \mathbf{P}[Q > b]}{h(b)} \leq \max\left[-\delta_- + \limsup_{b\to\infty}(\log \Delta(b))/h(b), \limsup_{b\to\infty} \frac{1}{h(b)} \log \sum_{t>\Delta(b)} \exp(v(t)\lambda_t^*(b/a(t))\right], \quad (13)$$

where

$$\delta_- = \liminf_{b\to 0} \frac{I(b)}{h(b)}, \quad \text{with} \quad I(b) = \inf_{t>0} v(t)\lambda_t^*(b/a(t)). \quad (14)$$

(ii) If, furthermore, Δ can be chosen such that

$$\lim_{t\to\infty} \frac{\log \Delta(b)}{h(b)} = 0, \quad (15)$$

and for all $\varepsilon_1 > 0$ and $\varepsilon_2 > 0$ sufficiently small

$$\lim_{\varepsilon\to 0} \lim_{t\to\infty} \frac{1}{h(b)} \log \sum_{t>\Delta(b)} e^{-\varepsilon_1 v(t)(\lambda^*(0)-\varepsilon_2)} \leq -\delta_-, \quad (16)$$

then

$$\limsup_{b\to\infty} \frac{1}{h(b)} \log \mathbf{P}[Q > b] \leq -\delta_-. \quad (17)$$

Proof of Theorem 2. (i) For any $t_0 > 0$, by using (5) to decompose $\{Q > b\} = \cup_{t>1}\{W_t > b\}$ and applying Chebychev's inequality on each component to get $P[W_t > b] \leq e^{-v(t)b\theta/a(t)}\mathbf{E}[\exp(\theta v(t)W_t/a(t))]$ for any $\theta > 0$, and so taking the infimum over $\theta > 0$ taking the infimum over $\theta > 0$ (which by the assumption on λ_t is equivalent to taking the infimum over all $\theta \in \mathbf{R}$) we get

$$\mathbf{P}[W_t > b] \leq \exp(-v(t)\lambda_t^*(b/a(t))). \quad (18)$$

Hence

$$\mathbf{P}[Q > b] \leq \Delta(b) \max_{t\leq\Delta(b)} \mathbf{P}[W_t > b] + \sum_{t>\Delta(b)} \mathbf{P}[W_t > b] \quad (19)$$

$$\leq \Delta(b)e^{-I(b)} + R(b), \quad \text{where} \quad R(b) = \sum_{t>\Delta(b)} e^{-v(t)\lambda_t^*(b/a(t)}. \quad (20)$$

The result the follows by applying the principle of the largest term (see [12]). The proof of (ii) follows trivially. ■

It is worth restating that Theorem 2 (ii) follows from (i) only when the tailsum term in (13) can be neglected w.r.t. δ_-. Two questions naturally arise from the preceding results. The first question is what are examples of processes that satisfy the hypothesis. The second is when are δ_- and δ_+ equal so that the bounds (6) are tight. All the examples considered in [8] satisfy the hypotheses of Theorem 1. What is new here are the growth conditions (15) and (16): they replace stronger conditions in [8], and as we shall see in the next section, admit a wider class of examples. Here we remark that $\delta_- = \delta_+$ in the examples of [8]. This includes, for instance, asymptotically self-similar processes with Hurst parameter $H \in [1/2, 1)$. In this case we should take $\tilde{a} = 1$, $\tilde{v} = 2 - 2H$, and Theorem 2(ii) follows by choosing $\Delta(b) = b \log b$.

Before leaving the general case, we prove the following inequality between δ_+ and δ_- which holds even in the absence of the lower bound in (6).

Theorem 3 *Assume for scaling functions v, a, h that (10) is satisfied for some f and that λ_t is pointwise convergent to λ. Then*

$$\delta_- = \liminf_{b\to\infty} \frac{I(b)}{h(b)} \leq \limsup_{b\to\infty} \frac{I(b)}{h(b)} \leq \delta_+ = \inf_{x>0} \frac{\lambda^*(x)}{f(x)}. \tag{21}$$

The proof uses the following Lemma, which is proved in [7].

Lemma 1 *Let χ_n be a sequence of convex functions on $\mathbf{R}$. If for some function χ, $\lim_{n\to\infty} \chi_n = \chi$ pointwise on the interior of the effective domain of χ, then $\lim_{n\to\infty} \chi_n^* = \chi^*$ pointwise on the interior of the effective domain of χ^*.*

Proof of Theorem 3. Only the second inequality has to be proved: the rest is definitions. Setting the dummy variable in the variational expression (14) for I to be $c = b/a(t)$, then any $x > 0$ we have for any $b, c > 0$ $I(b)/h(b) \leq f(b/c))\lambda^*_{a^{-1}(b/c)}(c)/h(b)$. Taking $b \to \infty$ for fixed c, $\limsup_{b\to\infty} I(b)/h(b) \leq \lambda^*(c)/f(c)$, for any c in the interior of the effective domain of λ^*, by Lemma 1 and (10). The result follows by taking the infimum over all $c > 0$ by virtue of (12) ■.

4 Applications to an $M/G/\infty$ arrival process.

Parulekar and Makowski [16] have determined the CGF for A_t for the discrete version of the $M/G/\infty$ process for classes of G and have used it the result of [8] to determine the tail asymptotics as in (6) and (17) with $\delta_+ = \delta_-$ for a number of models in which G is short tailed. In particular, in determining scaling functions a, v and appropriate for the CGF to exist, they take $a(t) = t$, and then derive the scaling function v directly from the model in question:

$$v(t) = -\log G_e^c(t)/dt \tag{22}$$

where G_e is the stationary excess distribution for the service time.

However, the long tailed case has features which take it out of the domain of Theorem 1 and (sometimes) Theorem 2. One difficulty is with the lower bound and stems from the form of the CGF λ. For $v(t) = o(t)$ it is shown in [17] that:

Theorem 4 *Assume $v(t) = o(t)$ with $v(t)/t, t = 1, 2, \ldots\}$ eventually monotone decreasing. Assume that there exists a mapping $\Gamma : \mathbf{N} \to \mathbf{N}$ such that (i) $\Gamma(t) < t$ for all t, (ii)*

$\lim_{t\to\infty} v(t)\Gamma(t)/t = \infty$ *and (iii)* $\lim_{t\to\infty} v(t)\Gamma(t)/(t\,v(\Gamma(t))) = 0$. *Then the limiting* CGF $\lambda(\theta) = \lim_{t\to\infty}(1/v(t))\log \mathbf{E}[\exp(v(t)W_t/t)]$ *exists and is given by*

$$\lambda(\theta) = \begin{cases} -\theta\phi & \text{if} \quad \theta < 1 \\ +\infty & \text{if} \quad \theta > 1 \end{cases}, \tag{23}$$

with $\lambda(1)$ *determined by further detail of the model in question.*

Now, this λ is not essentially smooth since it fails to be steep: $|\lambda'(\theta)|$ does not approach ∞ as θ approaches the boundary of its effective domain, namely 1. So the LD lower bound cannot be established by Theorem 1, However, things are even worse. Clearly, whatever the value of $\lambda(1) \geq \lim_{x\nearrow 1} \lambda(x)$ we have

$$\lambda^*(x) = \begin{cases} +\infty & \text{if} \quad x < -\phi \\ x + \phi & \text{if} \quad x > -\phi \end{cases}, \tag{24}$$

Thus, the only exposed point of λ^* (see Section 2.3 of [6] for terminology) is $-\phi$ and so the large deviation lower bound for the pair $(W_t/t, v(t))$ obtained through the Gärtner-Ellis theorem is trivial: for open sets G not containing $-\phi$: $\liminf_{t\to\infty} \log \mathbf{P}[W_t/t \in G]/v(t) \geq -\infty$. This is not to say that there is no non-trivial LDP, just that the Gärtner-Ellis theorem cannot be used to prove it. (See Remark (d) on page 46 of [6] for an example of a process with an LDP which also cannot be derived from this Theorem).

We now turn to the upper bound. As has been pointed out in [16], the hypotheses used in the proof in [8] of the LD upper bound in (6) are violated when $v(t) \sim \log t$. Unfortunately this precludes applying the theorems to the case of long tailed G. For example, when G is Pareto i.e.

$$\lim_{t\to\infty} \frac{G^c(t)}{t^{-\alpha}} = 1, \tag{25}$$

for some $\alpha > 1$, then

$$\lim_{t\to\infty} \frac{v(t)}{(\alpha - 1)\log t} = 1. \tag{26}$$

But we are able to do better on the basis of Theorem 2.

Theorem 5 *Consider the* $M/G/\infty$ *process when* G *has Pareto tail of power* $\alpha \in (0,1)$ *and suppose that* $(\alpha - 1)\phi > 1$. *Then* $\delta_- = \delta_+$ *and*

$$\limsup_{b\to\infty} \frac{\mathbf{P}[Q > b]}{\log b} \leq 1 - (\alpha - 1)\phi. \tag{27}$$

Proof of Theorem 5. To recap, we take $a(t) = t$, $v(t) = h(t) = (\alpha - 1)\log t$ and hence $f(x) = 1$. To obtain convergence of λ_t we apply Theorem 4 taking, for instance,

$$\Gamma(t) = \lfloor t/(1 + \log(1 + \log t))\rfloor, \tag{28}$$

Since $a(t) = t$, $\lambda_t'(0) = t^{-1}\mathbf{E}A_t - s = -\phi < 0$ for all t so that assumption (i) of Theorem 2 is satisfied. It is easy to see that λ_t^* is increasing on $\mathbf{R}_+$ as a consequence. Next observe that since $f(x) = 1$ and λ^* is increasing we have $\delta_+ = \phi = \lambda^*(0)$.

Since 0 lies in the interior of the domain of λ^*, we have by Lemmas 1 and the increasing property of λ_t^* on $\mathbf{R}_+$, that, for any $\varepsilon > 0$ $\lambda_t^*(b/a(t)) \geq \lambda^*(0) - \varepsilon = \delta_+ - \varepsilon$ for t sufficiently

large. So for b sufficiently large, then in (20) we have $R(b) \leq ((\alpha-1)\phi-1)^{-1}\Delta(b)^{1-(\alpha-1)\phi}$. Thus by Theorem 2(i)

$$\limsup_{b\to\infty} \frac{\log \mathbf{P}[Q > b]}{h(b)} \leq \max[\beta - \delta_-, \beta(1-(\alpha-1)\delta_+)], \text{ where } \beta = \limsup_{b\to\infty} \frac{\log \Delta(b)}{h(b)}. \quad (29)$$

We now show that δ_- and δ_+ are equal: then the upper bound is minimized by choosing $\Delta(b)$ such that $\beta = 1/(\alpha-1)$ and (27) obtains. By Theorem 3, $\delta_+ \geq \delta_-$, so we need only establish the reverse inequality.

We are free to replace $v(t)$ and $h(t)$ by $(\alpha-1)\log t$ for t greater than some t_0, forming a sequence to which they are asymptotically equivalent. We make the change of variable $c = b/a(t)$ and set $f_b(c) = (\log b)/\log(b/c)$. Then $I(b)/h(b) = \inf_{c>0} \lambda^*_{b/c}(c)/f_b(c) = \lambda^*_{b/c_b}(c_b)/f_b(c_b)$, where the infimum is achieved at c_b. (If the infimum is not achieved, use instead for any $\varepsilon > 0$ a point c_b where the infimum is achieved to within ε, then take $\varepsilon \to 0$ at the end). Then $\liminf_{b\to\infty} I(b)/h(b) = \lim_{b\to\infty} \lambda^*_{b/c_b}(c_b)/f_b(c_b)$ along some subsequence, which we also denote by (c_b). Now (c_b) must have a subsubsequence (which we also denote by (c_b)) with one of the following properties. Either c_b is eventually bounded; or $\lim_{b\to\infty} c_b = \infty$.

In the first case, $c_b < K$ for some $K > 0$ so that $\lim_{b\to\infty} 1/f_b(c_b) \geq 1$ while since λ^* is non-decreasing $\delta_- = \lim_{b\to\infty} \lambda^*_{a^{-1}(b/c_b)}(c_b) \geq \lim_{b\to\infty} \lambda^*_{a^{-1}(b/c_b)}(0) = \lambda^*(0) = \delta_+$ and we are done.

In the case $c_b \to \infty$, either b/c_b is bounded, or it diverges. If it is bounded then so is $t_b = a^{-1}(b/c_b)$, and so for sufficiently large b on the subsubsequence $I(b)/h(b) \geq k_1 \max_{1\leq t\leq t_0} \lambda^*_t(k_2 b)/\log b$, for some constants t_0, k_1, k_2. Clearly λ_t is finite and differentiable on $\mathbf{R}$, so that λ^*_t is strictly convex, and also $x \mapsto \lambda^*_t(x)$ is increasing when $x > s$. Hence for some $k_3 > 0$ $\lambda^*_t(x) > k_3 x$ for large enough x, and so $I(b)/h(b)$ diverges as $b \to \infty$. This contradicts the finiteness of δ_+: there is no such subsequence (c_b).

Finally, we treat the case that c_b and b/c_b are divergent. Then for any $K > 0$, $b > Kc_b$ eventually. Since $b \to 1/f_b(c)$ is increasing we get $1/f_b(c_b) \geq 1 - 1/\log K$ and since K is arbitrary $\liminf_{b\to\infty} 1/f_b(c_b) \geq 1$. Finally $\lambda^*_{a^{-1}(b/c_b)}(c_b) \geq \lambda^*_{a^{-1}(b/c_b)}(0) \to \lambda^*(0)$ as $b \to \infty$ and hence $b/c_b \to \infty$. Thus along the subsubsequence $\lim_{b\to\infty} I(b)/h(b) \geq \lambda^*(0) = \delta_+$ as required. ■

5 Asymptotics for large arrival rates at fixed load.

We now turn to our second large deviation principle, in which the arrival rate is scaled as rL and the service rate as sL where L grows to infinity. Consider first a slightly more general setup in which for each $L > 0$ we have a arrival process $A^L = (A^L_t : t \geq 0)$ served at rate sL. Define the corresponding excess workload process $W^L_t = A^L_t - sLt$ and queue length $Q^L_t = \sup_{t\geq 0} W^L_t$. Finally define the family of CGF's

$$\lambda^L_t(\theta) = (v(t)L)^{-1} \log \mathbf{E}[e^{\theta v(t) W^L_t/t}], \quad (30)$$

for some scaling functions $v(t)$. (Here $a(t) = t$).

Theorem 6 *Assume*

(i) For each $\theta \in \mathbf{R}$, the limit

$$\lambda_t(\theta) = \lim_{L\to\infty} \lambda^L_t(\theta) \quad (31)$$

exists as an extended real number, uniformly in t sufficiently large, and each λ_t is essentially smooth; or

(i') W_t^L *is an* L*-fold superposition of i.i.d. processes* W_t*, so that* $\lambda_t^L = \lambda_t$*.*

(ii) For some $\theta, \varepsilon > 0$*,* $\lambda_t(\theta) < -\varepsilon$ *for all* t*.*

(iii) For all $\varepsilon > 0$ $\lim_{t_0 \to \infty} \limsup_{L \to \infty} L^{-1} \log \sum_{t > t_0} e^{-\varepsilon L v(t)} = -\infty$*.*

Then with $I(b)$ *as in (14)*

$$\lim_{L \to \infty} L^{-1} \log \mathbf{P}[Q^L > Lb] = -I(b). \tag{32}$$

This theorem was proved under stronger conditions in [7]. Given the uniform convergence in (i), (ii) follows easily if λ_t converges to some λ satisfying Hypothesis (ii) of Theorem 1, at least on the interior of the effective domain of the latter. With an additional technical condition the result extends to continuous time. The canonical example of is when A_t^L is a homogeneous L-fold superposition of an arrival process A_t. This also applies to models with Poissonian arrivals, such as $M/G/\infty$, since scaling the arrival rate by $L \in \mathbf{N}$ in our $M/G/\infty$ models gives the same distribution for A_t^L as taking an L-fold superposition of A_t. We call I the *shape function*: it determines the shape of the log loss curve, asymptotically for large L. Comparing (32) with (14) we see that δ_- is a lower bound for the asymptotic decay rate (in b) of the loss ratio in both the single source and many source cases (when the appropriate large deviation theorems hold).

Most importantly for our present purpose, the hypotheses of Theorem 6 in some sense easier to satisfy than those of Theorem 2. One reason for this is that the transient CGF λ_t is typically essentially smooth in applications, whereas, as we have seen the previous section, the limit λ does not always have this property. The smoothness property plays different roles in each context. In Theorem 1 is is used to establish a temporal LDP for $(W^t/t, v(t))$ as $t \to \infty$. In Theorem 6 it is used to establish the spatial LDP for $(A_t^L/L - st, L)$ at each fixed t: this determines how quickly the Law of Large Numbers is obeyed as $L \to \infty$. This is proved using Gärtner-Ellis, or simply Cramer's theorem if the arrivals are i.i.d. superpositions. Also one can show that (15) and (16) in Theorem 2 imply hypothesis (iii) in Theorem 6.

Theorem 7 *Consider the* $M/G/\infty$ *arrival process of intensity* rL*, served at rate* sL*, where* G *is Pareto with power* $\alpha > 1$*. Then the associated workload processes* W_t^L *satisfy the hypotheses of Theorem 6 and (3) holds as* $\mathbf{N} \ni L \to \infty$*.*

Proof of Theorem 7. We establish the hypotheses of Theorem 6. For (i'), use the remark above that A_t^L has the distribution of an i.i.d. superposition. For (ii) we have seen in Theorem 5 the convergence of λ_t to λ when $v(t) = (\alpha - 1) \log t$. It is not difficult to see that for each t, λ_t is everywhere defined and hence differentiable. Since A_t is stationary and $a(t) = t$ we see that $\lambda_t'(0) = t^{-1}\mathbf{E}A_t - s = \lambda'(0) < 0$, the last inequality since λ is convex and (say) $\lambda(1/2) < 0$. Thus since λ_t is differentiable $\lambda_t(\theta_t) < 0$ for some $\theta_t > 0, t \in \mathbf{N}$. For each t let θ_t' be the supremum of such θ_t. But $\lim_{t \to \infty} \theta_t' = 0$ would contradict $\lambda(1/2) < 0$ and so (ii) holds. For (iii), by a similar argument to that used in Theorem 5, then for $\varepsilon' > \varepsilon > 1/L$ and sufficiently large t_0:

$$L^{-1} \log \sum_{t > t_0} \exp(-\varepsilon L v(t)) \leq -\varepsilon' t_0 (\alpha - 1) \tag{33}$$

from which (iii) follows. Thus the LDP (32) holds, and the asymptotic from of the shape function follows from Theorem 3 and the fact, established during the proof of Theorem 5 that $\delta_+ = \delta_-$. ■

We conclude by reviewing the behavior of the corresponding $M/G/1$ model with the same service time distribution G. Since the arrivals are Poissonian A_t^L is equal in distribution to A_{Lt} and hence

$$Q^L = \sup_{t>0}(A_t^L - cLt) \stackrel{d}{=} \sup_{t>0}(A_{Lt} - cLt) = \sup_{t\geq 0}(A_t - ct) = Q; \tag{34}$$

Thus $\mathbf{P}[Q^L > Lb] \approx k(bL)^{-(\alpha-1)}$ for some $k > 0$ and large L. This decays only algebraically in L, rather than exponentially as for $M/G/\infty$.

References

[1] V. Anantharam (1995). On the Sojourn Time of Sessions at an ATM Buffer with Long-Range Dependent Input Traffic, *Proceedings of the 34th IEEE Conference on Decision and Control, December 1995.*

[2] N.H. Bingham, C.M. Goldie and J.L. Teugels (1987). *Regular Variation*, Encyclopedia of Mathematics and its Applications. Vol 27. Cambridge University Press, Cambridge.

[3] A.A. Borovkov (1984). *Asymptotic Methods in Queueing Theory.* Wiley, Chichester.

[4] C.-S. Chang (1994). Stability, queue length and delay of deterministic and stochastic queueing networks. *IEEE Trans. on Automatic Control*, **39**:913–931.

[5] G.L. Choudhury and W. Whitt (1996). Long-tail buffe-content distributions in broadband networks. Preprint, AT&T Laboratories.

[6] A. Dembo and O. Zeitouni (1993). *Large Deviation Techniques and Applications.* Jones and Bartlett, Boston-London.

[7] N.G. Duffield (1996). Economies of scale in queues with sources having power-law large deviation scalings. *J. Appl. Prob.*, **33:** 840–857.

[8] N.G. Duffield and N. O'Connell (1995). Large deviations and overflow probabilities for the general single-server queue, with applications, *Math. Proc. Cam. Phil. Soc.*, **118**:363–374.

[9] N.G. Duffield and W. Whitt (1996). Recovery from rare congestion events in multi-server systems, Preprint, AT&T Laboratories.

[10] P.W. Glynn and W. Whitt (1994). Logarithmic asymptotics for steady-state tail probabilities in a single-server queue. In: *Studies in Applied Probability* Eds. J. Galambos and J. Gani, *Journal of Applied Probability, Special Volume* **31A** 131–159

[11] G. Kesidis, J. Walrand and C.S. Chang (1993). Effective bandwidths for multiclass Markov fluids and other ATM Sources. *IEEE/ACM Trans. Networking*, **1**:424-428.

[12] J.T. Lewis and C.E. Pfister (1995). Thermodynamic probability theory: some aspects fo large deviations. *Russian Mathematical Surveys*, **50**:2 47-83.

[13] N. Likhanov, B. Tsybakov and N.D. Georganas (1995). Analysis of an ATM buffer with self-similar ("fractal") input traffic, *Proceedings of Infocom95,* Boston, April 1995, pp. 985–992.

[14] B.B. Mandelbrot and J.W. Van Ness (1968). Fractional Brownian motions, fractional noises and applications. *SIAM Review*, **10**:422–437.

[15] I. Norros (1994). A storage model with self-similar input. *Queueing Systems*, **16**:387–396

[16] M. Parulekar and A. Makowski (1996). Tail probabilities for a multiplexer with self-similar traffic. *Proc. IEEE INFOCOM'96, San Francisco, March 26-28, 1996*, pp1452–1459.

[17] M. Parulekar and A. Makowski (1996). Tail probabilities for $M/G/\infty$ input processes(I): preliminary asymptotics. Preprint, University of Maryland.

[18] V. Paxson and S. Floyd (1995). Wide-area traffic: the failure of Poisson modelling. *IEEE/ACM Transactions on Networking*, **3**:226-244.

[19] R.T. Rockafellar (1970) *Convex Analysis.* Princeton University Press, Princeton.

On the typical path to hitting a rare set

Tim Zajic
Department of Industrial Engineering and Operations Research
Columbia University, New York
and
IBM T.J. Watson Research Center
Yorktown Heights, New York

Abstract

Given a sequence of $\mathbb{R}^d$-valued random variables and an open convex subset of $\mathbb{R}^d$, A, asymptotic results are derived describing the manner in which the sequence behaves, given that the time to hit MA was finite, as M increases through the positive integers. An application of the result is briefly discussed.

1 Introduction and statement of result.

Given a sequence of $\mathbb{R}^d$-valued random variables, $\{Y_n\}_{n\geq 1}$, and a set $A \subset \mathbb{R}^d$, let $T = \inf\{n \geq 1 \ : \ Y_n \in MA\}$, where M is a positive integer and the dependence of T on M is suppressed. Interest in the behaviour of T for large M appears, for example, in risk theory (cf. [2, 8] and references therein). Nyrhinen [8] considered the behaviour of T for large M in case $d = 1$, $A = [1, \infty)$ and the sequence of generating functions corresponding to the sequence $\{Y_n\}_{n\geq 1}$ satisfies appropriate conditions. In addition, for $r > 0$ fixed, the typical behaviour for large M, made precise in Section 2, of the continuous time process $\{Y_{\lceil \alpha M \rceil}/M \ : \ \alpha \in [0, r]\}$, conditioned on the event $T < \infty$, is studied.

Collamore [2], again under assumptions placed on the generating functions corresponding to $\{Y_n\}_{n\geq 1}$, considers the case $d \geq 1$. In addition to studying the behaviour of T for M large, the behaviour of Y_T is considered. The interest here is in describing, in case $d > 1$, where the sequence enters the set A. However, the consideration of the

continuous time process, analogous to that studied in [8] mentioned in the previous paragraph, is not taken up.

We study here the continuous time process considered in [8] in the setting of [2]; namely, we allow for $d > 1$. In [8], an important ingredient in the study of the continuous time process is the behaviour of T for large M. In case $d > 1$, it is important to consider also the behaviour of Y_T; i.e. where the set A is entered.

In Section 2 we provide relevant background results, the result we wish to prove and examples of sequences which satisfy our assumptions. Section 3 is devoted to the proof of our result, which is based on that of [8, Theorem 6]. We close in Section 4 by mentioning an application of our result which is currently under investigation.

2 Preliminaries, statement of result and examples.

Let $\{Y_n\}_{n\geq 1}$ be a sequence of $\mathbb{R}^d$ valued random variables and $A \subset \mathbb{R}^d$ an open convex set. Let

$$\Lambda(\alpha) = \limsup_{n\to\infty} \frac{1}{n} \log E[e^{\langle \alpha, Y_n \rangle}], \quad \forall\, \alpha \in \mathbb{R}^d .$$

Assuming the sequence $\{Y_n\}_{n\geq 1}$ satisfies a large deviations principle in $\mathbb{R}^d$ with convex lower-semicontinuous rate function I (cf. Section 1.2 of [4]), let

$$I^*(x) = \sup_{\alpha\in\mathbb{R}^d} \{\langle \alpha, x\rangle - I(\alpha)\}, \quad \forall\, x \in \mathbb{R}^d$$

$$L_0 I^* = \{x : I^*(x) \leq 0\} \quad \text{and} \quad L_0 I = \{x : I(x) \leq 0\} ,$$

with $L_0 I$ assumed to be non-empty. Continuing, let $D = \{\lambda x : \lambda > 0, I(x) < \infty\}$; that is, the positive cone generated by the domain of I, $\mathrm{dom}(I)$. Finally, for M and N positive integers, let $T_N = \inf\{n \geq N : Y_n \in MA\}$, the dependence on M being suppressed. We recall the following conditions from Collamore [2]:

(H1) $\Lambda(\alpha) < \infty$ for all $\alpha \in L_0 I^*$.

(H2) For some $\delta > 0$, $A \cap \{\lambda x : \lambda > 0, x \in C_\delta\} = \emptyset$, where $C_\delta = \{x : \inf_{y\in L_0 I} \|x - y\| < \delta\}$ in case the level sets of I are compact and $C_\delta = \{x : \|x - y\| < (1 + \|y\|)\delta$, for some $y \in L_0\}$ otherwise.

The following theorem appears as Theorem 2.1 in [2].

Theorem 1 *Suppose (H-1) and (H-2) hold and that A has a non-empty intersection with the relative interior of D. Then, for N large enough,*

$$\lim_{M\to\infty} \frac{1}{M} \log P(T_N < \infty) = -\inf_{x\in A} \sup_{\alpha\in L_0 I^*} \langle \alpha, x\rangle .$$

In case

$$\sup_{n\geq 1}\frac{1}{n}\log E[e^{\langle\alpha,Y_n\rangle}]<\infty \text{ for all } \alpha\in L_0I^*\ ,$$

we may take $N=1$ in the statement of Theorem 1. Indeed, this last statement holds valid for all theorems stated in this section.

Assume further that

(H3) Λ exists as a limit, $I^*=\Lambda$ and $\Lambda(\cdot)$ is differentiable on L_0I^*.

We remark that (H3) implies that there exists a unique $\hat{x}$ in the closure of A such that $\sup_{\alpha\in L_0I^*}\langle\alpha,\hat{x}\rangle=\inf_{x\in A}\sup_{\alpha\in L_0I^*}\langle\alpha,x\rangle$ (see [1]). The proofs of the following two theorems may be carried out using the same method of proof used in proving the results appearing in Section 5 of [2].

Theorem 2 *Assume in addition to the conditions of Theorem 1 that (H3) holds. For N large enough, for every $\epsilon>0$ there exists a constant $K(\epsilon)>0$ such that*

$$P(\|\frac{Y_{T_N}}{M}-\hat{x}\|>\epsilon|T_N<\infty)\leq\exp(-MK(\epsilon)) \text{ for large } M.$$

Theorem 3 *Assume in addition to the conditions of Theorem 1 that (H3) holds. For N large enough, there exists $\rho>0$ such that for every $\epsilon>0$ there exists a constant $L(\epsilon)>0$ such that*

$$P(|\frac{T_N}{M}-\rho|>\epsilon|T_N<\infty)\leq\exp(-ML(\epsilon)) \text{ for large } M.$$

Given these results we shall prove the analogue of Theorem 6 of Nyrhinen [8]. We first present some definitions before stating our theorem.

Let $w=\inf_{x\in A}\sup_{\alpha\in L_0I^*}\langle\alpha,x\rangle$. By the proof of Lemma 3.2 of [2], there exists an $\hat{\alpha}\in L_0I^*$ satisfying $\langle\hat{\alpha},\hat{x}\rangle=w$. Fix $i\in\{1,\ldots,d\}$ and define, for $t\in\mathbb{R}$ and $\beta,\nu\geq 0$

$$C_M^i(t,\hat{\alpha};\beta,\nu)=M^{-1}\log E[e^{tY^i_{\lceil\beta M\rceil}+\langle\hat{\alpha},Y_{\lceil\nu M\rceil}\rangle}]\ .$$

Define also, for $\alpha\geq 0$ and $\delta>0$, $H^i_{\alpha,\delta}:\mathbb{R}\to\mathbb{R}\cup\{\pm\infty\}$ by

$$H^i_{\alpha,\delta}(t)=\limsup_{M\to\infty}\sup_{\beta\in B(\alpha,\delta)}C^i_M(t,\hat{\alpha};\beta,\rho)$$

and let

$$H^i_\alpha(t)=\lim_{\delta\to 0^+}H^i_{\alpha,\delta}(t)\ .$$

The proof of Lemma 5 of [8] shows that $H^i_\alpha(\cdot)$ is proper convex and finite in a neighbourhood of the origin. In case $H^{i\prime}_\alpha(0)$ exists we denote it by $p^i(\alpha)$. Finally, define, for $\alpha>0$, $Y_M(\alpha)=Y_{\lceil\alpha M\rceil}$ and for any vector Y denote by Y^i the i-th component of Y.

Theorem 4 *Suppose in addition to the conditions of Theorem 3 that, for some $r > 0$, $H_\alpha^{i\prime}(0)$ exists for every $\alpha \in [0, r]$. Then, for N large enough, for each $\epsilon > 0$ there exist constants $k_\epsilon > 0$ and M_ϵ such that*

$$P\Big(\sup_{0<\alpha\le r} \Big|\frac{Y_M^i(\alpha)}{M} - p^i(\alpha)\Big| \ge \epsilon | T_N < \infty\Big) \le e^{-k_\epsilon M}$$

for every $M > M_\epsilon$.

We present the proof of Theorem 4 in the following section. We end this section with three examples of sequences $\{Y_n\}_{n\ge 1}$ for which (H1) and (H3) hold. We note that in each case I has compact level sets and that $L_0 I = \nabla\Lambda(0)$ (cf. Theorem VII.2.1 of [5]). Hence, given A, it becomes a simple matter to check (H2). Furthermore, as Λ is obtained as a finite limit in each case it follows that $I^* = \Lambda$ (see, for example, [4, Theorem 4.5.10]). Finally, in each of the examples the function $p^i(\cdot)$ is seen to consist of two linear segments joined at the point ρ. In stating the examples we denote by e_i the vector consisting of a 1 in the i-th position and zeros in all others.

Example 1 Let $\{X_i\}$ be a stationary sequence of m-dependent random variables (that is, X_i and X_{i+j} are independent for $i \ge 1$ and $j \ge m$) and $Y_n = \sum_{i=1}^n X_i$. If $E[\exp(\langle\alpha, X_i\rangle)] < \infty$ for all $\alpha \in \mathbb{R}^d$, then

$$\Lambda(\alpha) = \frac{1}{m} \log E[e^{\langle\alpha, \sum_{i=1}^m X_i\rangle}]$$

and

$$H_\alpha^i(t) = \begin{cases} \alpha\Lambda(t\cdot e_i + \hat{\alpha}) & \text{if } \rho \ge \alpha \\ \rho\Lambda(t\cdot e_i + \hat{\alpha}) + (\alpha - \rho)\Lambda(t\cdot e_i) & \text{otherwise .} \end{cases} \tag{1}$$

Furthermore, Λ is differentiable and $\nabla\Lambda(0) = EX_1$. That $\{Y_n\}$ satisfies a large deviations principle with good convex rate function $I = \Lambda^*$ follows from the Gärtner-Ellis Theorem (cf. Theorem 2.3.6 of [4]). As knowledge of $\mathrm{dom}(I)$ is necessary in applying Theorem 4, we provide two means of verifying when $I(x) < \infty$ for $x \in \mathbb{R}^d$. First, it follows from [3, Exercise 2.2.39 (a)] that, for fixed $x \in \mathbb{R}^d$, a sufficient condition for $I(x) < \infty$ is that the distribution of $\sum_{i=1}^m X_i$ possess a density which is continuous and positive at x. Secondly, suppose that $P(\sum_{i=1}^m X_i = y) = p > 0$ for some $y \in \mathbb{R}^d$. It then follows that, for $x \in \mathbb{R}^d$,

$$\langle\lambda, x\rangle - \frac{1}{m}\log E[\exp^{\langle\lambda, \sum_{i=1}^m X_i\rangle}] \le \langle\lambda, x\rangle - \frac{1}{m}\log(p\exp^{\langle\lambda, y\rangle})$$

$$= \langle\lambda, x\rangle - \langle\lambda, \frac{y}{m}\rangle - \frac{1}{m}\log p\,.$$

From this, we see that $I(y/m) \le -\log p/m$.

Example 2 Suppose $\{(X_i, S_i)\}$ is a Markov-additive process satisfying the uniform recurrence condition (R) of [7] and $E[\exp(\langle \alpha, x \rangle)] < \infty$ for all $\alpha \in \mathbb{R}^d$, where the expectation is with respect to the measure $\nu(\cdot)$ appearing in condition (R). We let $\{Y_n\} = \{S_n\}$. It follows from [7] that $\Lambda(\alpha)$ exists for all $\alpha \in \mathbb{R}^d$ and from [3] that $H^i_\alpha(\cdot)$ is given as in (1). Furthermore, Λ is differentiable and $\nabla\Lambda(0) = E_\pi S_1$, where π is an invariant measure for the underlying Markov chain (see [7]). That $\{Y_n\}$ satisfies a large deviations principle with good convex rate function $I = \Lambda^*$ follows from the results of [7]. Sufficient conditions for $I(x) < \infty$ may be phrased in terms of $\nu(\cdot)$. Recall (cf. [7, Lemma 3.1]) that

$$\log \int_{\mathbb{R}^d} e^{\langle \alpha, x \rangle} d\nu(E \times dx) + \log a \leq \Lambda(\alpha) \, ,$$

where $a > 0$. Given this, a sufficient condition follows as in [3, Exercise 2.2.39 (a)] and is that $\nu(E \times \cdot)$ possess a density which is positive and continuous at x. Alternatively, in case $\nu(E \times y) = p > 0$ for some $y \in \mathbb{R}^d$, it is readily seen that $I(ay) \leq -\log(pa)$.

Example 3 Let $\{Z_i\}$ be an i.i.d. $\mathbb{R}^m$ valued sequence such that $E[\exp(\langle \alpha, Z_1 \rangle)] < \infty$ for all $\alpha \in \mathbb{R}^m$ and $\{C_i\}_{i=0}^\infty$ a sequence of $d \times m$ matrices such that the sum of the matrices converges. Setting $\xi_i = \sum_{i=0}^{n-1} C_i Z_{n-i}$, we consider $Y_n = \sum_{i=1}^n \xi_i$. Calculating as in [8], where the case with $d = 1$ is considered, we find that, with Λ^Z denoting the logarithmic moment generating function of Z_1,

$$\Lambda(\lambda) = \Lambda^Z(\langle \lambda, \sum_{i=0}^\infty C_i^{\text{col 1}} \rangle, \ldots, \langle \lambda, \sum_{i=0}^\infty C_i^{\text{col m}} \rangle) \, ,$$

where, for any matrix B, $B^{\text{col i}}$ denotes the i-th column of the matrix. With Λ thus defined, $H^i_\alpha(\cdot)$ is given as in (1). Furthermore, Λ is differentiable and $\nabla\Lambda(0) = (\sum_{i=0}^\infty C_i) E Z_1$. That $\{Y_n\}$ satisfies a large deviations principle with good convex rate function $I = \Lambda^*$ follows from the Gärtner-Ellis Theorem (cf. Theorem 2.3.6 of [4]). A sufficient condition for $I(x) < \infty$ is that the distribution of $(\sum_{i=0}^\infty C_i) Z_1$ possess a density which is continuous and positive at x. Alternatively, in case $P(Z_1 = y) = p > 0$ for some $y \in \mathbb{R}^d$, it follows that $I(\langle \sum_{i=0}^\infty C_i, y \rangle) \leq -\log p$.

3 Proof of Theorem 4.

The proof parallels that of Theorem 6 of [8]. We first show that for $\epsilon > 0$ given and $\alpha \in [0, r]$ fixed, we can find $\delta_\alpha > 0, k_\alpha > 0$ and M_0 such that

$$P(\sup_{\beta\in B_0(\alpha,\delta_\alpha)\cap(0,r]}|\frac{Y_M^i(\beta)}{M}-p^i(\beta)|\geq\epsilon|T_N<\infty)\leq e^{-k_\alpha M} \tag{2}$$

for every $M > M_0$, where N is large enough. Here, $B(\alpha,\delta_\alpha)$ denotes the open ball with center α and radius δ_α and $B_0(\alpha,\delta_\alpha)=B(\alpha,\delta_\alpha)\cap(0,\infty)$. Fix small $\epsilon'>0, \delta>\alpha\epsilon'/(\rho-\epsilon')$ and choose $\delta_\alpha>0$ such that

$$\{\beta\rho/\nu|\beta\in B_0(\alpha,\delta_\alpha),\nu\in\bar{B}(\rho,\epsilon')\}\subset B(\alpha,\delta)\,.$$

By Chebycheff's inequality, for $\beta,\nu\geq 0$ and $\tilde{\epsilon}>0$,

$$e^{M(C_M^i(t,\hat{\alpha};\beta,\nu)+\tilde{\epsilon}\|\hat{\alpha}\|)}\geq e^{wM}E[e^{tY^i_{\lceil\beta M\rceil}}1(Y_{\lceil\nu M\rceil}\in MA,\ \|Y_{\lceil\nu M\rceil}-M\hat{x}\|<\tilde{\epsilon}M)]\,.$$

This yields

$$\begin{aligned}&E[e^{tY^i_{\lceil\beta M\rceil}}1(T_N/M\in B(\rho,\epsilon'),\ \|Y_{T_N}-M\hat{x}\|<\tilde{\epsilon}M)]\\&\leq[(\rho+\epsilon')M]\sup_{\nu\in\bar{B}(\rho,\epsilon')}E[e^{tY^i_{\lceil\beta M\rceil}}1(Y_{\lceil\nu M\rceil}\in MA,\ \|Y_{\lceil\nu M\rceil}-M\hat{x}\|<M\tilde{\epsilon})]\\&\leq[(\rho+\epsilon')M]e^{-wM}\sup_{\nu\in\bar{B}(\rho,\epsilon')}e^{MC_M^i(t,\hat{\alpha};\beta,\nu)}e^{\tilde{\epsilon}M\|\hat{\alpha}\|}\,.\end{aligned}$$

With $M_\nu=\nu M/\rho$, note that

$$C_M^i(t,\hat{\alpha};\beta,\nu)=C^i_{M_\nu}(t,\hat{\alpha};\beta\rho/\nu,\rho)\nu/\rho\,.$$

This allows to write

$$\begin{aligned}&\sup_{\beta\in B_0(\alpha,\delta_\alpha)}M^{-1}\log E[e^{tY^i_{\lceil\beta M\rceil}}1(T_N/M\in B(\rho,\epsilon'),\ \|Y_{T_N}-M\hat{x}\|<\tilde{\epsilon}M)]\\&\leq -w+\sup_{\beta\in B_0(\alpha,\delta_\alpha),\nu\in\bar{B}(\rho,\epsilon')}C^i_{M_\nu}(t,\hat{\alpha};\beta\rho/\nu,\rho)\nu/\rho+\tilde{\epsilon}\|\hat{\alpha}\|+o(1)\\&\leq -w+\sup_{L\geq(\rho-\epsilon')M/\rho}\{\sup_{\beta\in B_0(\alpha,\delta),\nu\in\bar{B}(\rho,\epsilon')}C^i_L(t,\hat{\alpha};\beta,\rho)\nu/\rho\}+\tilde{\epsilon}\|\hat{\alpha}\|+o(1)\,.\end{aligned}$$

We now have

$$\begin{aligned}&\limsup_{M\to\infty}\sup_{\beta\in B_0(\alpha,\delta_\alpha)}M^{-1}\log E[e^{tY^i_{\lceil\beta M\rceil}}1(T_N/M\in B(\rho,\epsilon'),\ \|Y_{T_N}-M\hat{x}\|<\tilde{\epsilon}M)]\\&\leq -w+H^i_{\alpha,\delta}(t)+\epsilon'|H^i_{\alpha,\delta}(t)|/\rho+\tilde{\epsilon}\|\hat{\alpha}\|\,.\end{aligned}$$

By Theorems 1, 2 and 3, we have, for N large enough,

$$\begin{aligned}&\limsup_{M\to\infty}\sup_{\beta\in B_0(\alpha,\delta_\alpha)}M^{-1}\log E[e^{tY^i_{\lceil\beta M\rceil}}|T_N/M\in B(\rho,\epsilon'),\ \|Y_{T_N}-M\hat{x}\|<\tilde{\epsilon}M]\\&\leq H^i_{\alpha,\delta}(t)+\epsilon'|H^i_{\alpha,\delta}(t)|/\rho+\tilde{\epsilon}\|\hat{\alpha}\|\,.\end{aligned}$$

Given $\hat{\epsilon} > 0$, possibly decreasing $\epsilon', \tilde{\epsilon} > 0$, (2.9), (2.10) and Lemma 1 of [8] yield that, for some $k > 0$,

$$\sup_{\beta \in B_0(\alpha,\delta_\alpha)} P(\frac{Y^i_{\lceil \beta M \rceil}}{M} \geq H^{i'}_{\alpha,\delta}(0+) + \hat{\epsilon}/2 | T_N/M \in B(\rho,\epsilon'),\ \|Y_{T_N} - M\hat{x}\| < \tilde{\epsilon}M) \leq e^{-kM}$$

and

$$\sup_{\beta \in B_0(\alpha,\delta_\alpha)} P(\frac{Y^i_{\lceil \beta M \rceil}}{M} \leq H^{i'}_{\alpha,\delta}(0-) - \hat{\epsilon}/2 | T_N/M \in B(\rho,\epsilon'),\ \|Y_{T_N} - M\hat{x}\| < \tilde{\epsilon}M) \leq e^{-kM}$$

for $M > M_0$ and M_0 and N large enough. Furthermore, since, by Lemmas 1 and 5 of [8],

$$\lim_{\delta \to 0^+} H^{i'}_{\alpha,\delta}(0+) = \lim_{\delta \to 0^+} H^{i'}_{\alpha,\delta}(0-) = p^i(\alpha)\ ,$$

decreasing ϵ', δ and δ_α if necessary, we have that

$$\sup_{\beta \in B_0(\alpha,\delta_\alpha)} P(|\frac{Y^i_{\lceil \beta M \rceil}}{M} - p^i(\alpha)| \geq \hat{\epsilon} | T_N/M \in B(\rho,\epsilon'),\ \|Y_{T_N} - M\hat{x}\| < \tilde{\epsilon}M) \leq 2e^{-kM} \quad (3)$$

for $M > M_0$ and M_0 and N large enough. By Lemma 5 of [8], $p^i(\cdot)$ is continuous on $[0, r]$ and hence, assuming that

$$\sup_{\beta \in B_0(\alpha,\delta_\alpha) \cap (0,r]} |p^i(\beta) - p^i(\alpha)| \leq \hat{\epsilon}$$

we may write, for $\hat{\epsilon} \leq \epsilon/2$

$$P(\sup_{\beta \in B_0(\alpha,\delta_\alpha) \cap (0,r]} |\frac{Y^i_M(\beta)}{M} - p^i(\beta)| \geq \epsilon | T_N/M \in B(\rho,\epsilon'),\ \|Y_{T_N} - M\hat{x}\| < \tilde{\epsilon}M)$$

$$\leq (\alpha + \delta_\alpha)M \sup_{\beta \in B_0(\alpha,\delta_\alpha)} P(|\frac{Y^i_{\lceil \beta M \rceil}}{M} - p^i(\alpha)| \geq \hat{\epsilon} | T_N/M \in B(\rho,\epsilon'),\ \|Y_{T_N} - M\hat{x}\| < \tilde{\epsilon}M)\ .$$

Together with (3) it follows that

$$P(\sup_{\beta \in B_0(\alpha,\delta_\alpha) \cap (0,r]} |\frac{Y^i_M(\beta)}{M} - p^i(\beta)| \geq \epsilon | T_N/M \in B(\rho,\epsilon'),\ \|Y_{T_N} - M\hat{x}\| < \tilde{\epsilon}M) \leq e^{-k_2 M} \quad (4)$$

for some $k_2 > 0$ and for $M > M_0$ and M_0 and N large enough. Next, we note that

$$P(\sup_{\beta \in B_0(\alpha,\delta_\alpha) \cap (0,r]} |\frac{Y^i_M(\beta)}{M} - p^i(\beta)| \geq \epsilon | T_N < \infty)$$

$$\leq P(\sup_{\beta \in B_0(\alpha,\delta_\alpha) \cap (0,r]} |\frac{Y^i_M(\beta)}{M} - p^i(\beta)| \geq \epsilon | T_N/M \in B(\rho,\epsilon'),\ \|Y_{T_N} - M\hat{x}\| < \tilde{\epsilon}M)$$

$$+ P(T_N/M \in B(\rho,\epsilon'),\ \|Y_{T_N} - M\hat{x}\| < \tilde{\epsilon}M | T_N < \infty)\ .$$

Together with (4) and Theorems 2 and 3, (2) follows. The statement of the theorem now follows from the Heine-Borel theorem and a union bound. ■

4 An application of Theorem 4.

The desire for a result of the type stated in Theorem 4 arose during the investigation of a variance reduction technique for rare event simulation known as *splitting*. In Glasserman, et. al. [6] the performance of splitting was analyzed under appropriate assumptions on the process to be simulated and shown to be asymptotically optimal. Key to the analysis was the identification of the estimator with a branching process, allowing the application of results and methods from this area of study. The authors of [6] are currently continuing the investigation of splitting, one of the objectives being to broaden the class of processes considered in [6]. Thus far, Theorem 4 has been applied in deriving necessary conditions for asymptotic optimality of the splitting estimator in a setting more general than that considered in [6]. As the investigation is currently in progress we limit ourselves to these few words.

Acknowledgments: I would like to thank Jeffrey Collamore for making available a preprint of his work and for discussing his results. This work was made possible by NSF Grant DMS-9508709.

References

[1] J. Collamore, *Large Deviations Techniques for the Study of Hitting Probabilities into Rare Sets*, Ph.D. dissertation, University of Wisconsin-Madison, 1996.

[2] J. Collamore, Hitting probabilities and large deviations. To appear in Annals of Probability.

[3] A. Dembo and T. Zajic, Large deviations: from empirical mean and measure to partial sums process. *Stoch. Processes and Appl.*, 57 (1995) 191-224.

[4] A. Dembo and O. Zeitouni, *Large Deviations Techniques and Applications.* Boston: Jones and Bartlett Publishers, 1993.

[5] R. S. Ellis, *Entropy, Large Deviations, and Statistical Mechanics.* New York: Springer-Verlag, 1985.

[6] P. Glasserman, P. Heidelberger, P. Shahabuddin and T. Zajic, Multilevel splitting for estimating rare event probabilities. IBM Research Report RC 20478. Yorktown Heights, New York.

[7] I. Iscoe, P. Ney and E. Nummelin, Large deviations of uniformly recurrent Markov additive processes. *Adv. Appl. Math.* 6 (1985) 383-412.

[8] H. Nyrhinen, On the typical level crossing time and path. *Stoch. Processes and Appl.*, 58 (1995) 121-137.

NUMERICAL COMPUTATION OF LARGE DEVIATIONS EXPONENTS VIA SIMULATION

Peter W. Glynn*
Department of Engineering-Economic
Systems and Operations Research
Stanford University
Stanford, CA 94305-4023, USA

Abstract

Consider the problem of numerically computing the exponential rate at which the tail of the hitting time of a Markov chain to a given set decreases to zero. One approach involves computing the solution to an eigenvalue problem. In this paper, we use a regenerative representation for the exponential rate constant to construct an associated simulation-based estimator. A strong law and central limit theorem for the estimator are also presented.

Keywords

Large deviations, simulation, regeneration, eigenvalue problem, hitting time distribution, Markov chains.

1 Introduction

It has long been known that the hitting time distributions of Markov processes in both discrete and continuous time often exhibit exponential decay in their tail probabilities. However, the numerical computation of the corresponding decay rate constants has received less attention.

In this paper, we consider a simulation-based approach for computing such decay rate constants (referred to, in the sequel, as the "large deviations exponent"). Our approach depends upon a regenerative representation for such exponents that was implicit in the work of Ney and Nummelin (1987) on large deviations for additive functionals of Markov processes.

This paper is organized as follows. Section 2 sets up the basic mathematical framework in which our analysis will be conducted. In Section 3, we represent the large deviations exponent of interest in terms of a regenerative quantity, and provide some associated theory. Section 4 describes our estimator, and provides a law of large numbers and central limit theorem for our methodology.

*This research was supported by the Army Research Office under Contract No. DAAH04-94-G-0021

2 The Basic Framework

Let $X = (X_n : n \geq 0)$ be a (time-homogeneous) Markov chain living on a finite or countably infinite state space S, with one-step transition matrix $P = (P(x,y) : x,y \in S)$. For a given non-empty subset $A \subseteq S$ such that $A^c \neq \emptyset$, let

$$T = \inf\{n \geq 0 : X_n \in A\}$$

be the first hitting time of A. Our interest is in studying the rate at which the tail probabilities of T decrease to zero, given that the process is initially started in A^c.

Consider the assumption:

A1 The restriction of P to A^c, namely $B = (P(x,y) : x,y \in A^c)$, is irreducible.

For $x \in S$, let $P_x(\cdot) = P(\cdot | X_0 = x)$. Using elementary arguments, it is straightforward to establish the following result, due to Vere-Jones (1962) and Kingman (1963).

Theorem 1 *Under Assumption A1, there exists $\lambda \geq 0$ such that for each $x,y \in A^c$,*

$$\frac{1}{n} \log P_x(X_n = y, T > n) \to -\lambda \tag{1}$$

as $n \to \infty$.

Theorem 1 suggests the approximation

$$P_x(X_n = y, T > n) \approx \exp(-\lambda n) \tag{2}$$

for n large. (An important consequence of the theorem is that the exponential rate constant λ is independent of x and y.) This is, of course, a special type of "large deviations" asympototic. Our interest is in efficient numerical computation of the "large deviations exponent" λ.

Before proceeding, it should be noted that $A1$ does not generally imply either

$$\frac{1}{n} \log P_\mu(X_n = y, T > n) \to -\lambda$$

or that

$$\frac{1}{n} \log P_x(T > n) \to -\lambda$$

as $n \to \infty$. (Here, $P_\mu(\cdot)$ is the probability distribution of X, conditional on X_0 following distribution μ.) It is easy to construct non-pathological examples in which the large deviations exponents for $P_\mu(X_n = y, T > n)$ and $P_x(T > n)$ differ from λ. Even simple random walks provide such examples; see Kingman (1963) for details. (Of course, it is obvious that all such examples must involve chains in which $|A^c| = +\infty$.) Thus, it is evident that additional restrictions must be imposed on X in order to guarantee that these various large deviations exponents coincide.

3 The Regenerative Viewpoint

The regenerative structure of discrete state space Markov chains is commonly used to study the steady-state of such stochastic processes. Here, we consider the use of regeneration in studying the hitting time distribution introduced in Section 2.

For $x, y \in A^c$, let

$$a(n) = P_x(X_n = y, T > n).$$

Set $\tau = \inf\{n \geq 1 : X_n = x\}$ and note that the strong Markov property, applied at time τ, implies that

$$a(n) = P_x(X_n = y, T \wedge \tau > n) + \sum_{j=1}^{n} P_x(X_n = y, T > n, \tau = j) \quad (3)$$

$$= b(n) + \sum_{j=1}^{n} a(n-j)g(j), \quad (4)$$

where

$$b(n) = P_x(X_n = y, T \wedge \tau > n)$$
$$g(n) = P_x(\tau = n, T > \tau)$$

Consequently, $a = (a(n) : n \geq 0)$ satisfies a (defective) renewal equation. The standard approach to studying such equations is to invoke the following assumption:

A2 There exists $\theta \geq 0$ such that

$$\sum_{n=1}^{\infty} e^{\theta n} g(n) = 1$$

Assumption A2 may be rewritten in the form: There exists $\theta \geq 0$ such that

$$E_x[e^{\theta\tau}; T > \tau] = 1.$$

In any case, if we let $\tilde{a}(n) = e^{\theta n}a(n)$, $\tilde{b}(n) = e^{\theta n}b(n)$, $\tilde{g}(n) = e^{\theta n}g(n)$, we note that by multiplying through Equation 4 by $e^{\theta n}$ that we arrive at the following (proper) renewal equation:

$$\tilde{a}(n) = \tilde{b}(n) + \sum_{j=1}^{n} \tilde{a}(n-j)\tilde{g}(j) \quad (5)$$

Since $(\tilde{g}(n) : n \geq 1)$ is a probability mass function, conventional renewal theory may be applied to Equation 5. To do so, we require that:

A3 i) $(\tilde{g}(n) : n \geq 1)$ is an aperiodic sequence;
ii) $\sum_{n=1}^{\infty} n\tilde{g}(n) < \infty$.

Observe that Assumption A3 ii) may be re-expressed as requiring that

$$E_x[\tau e^{\theta\tau}; T > \tau] < \infty,$$

where θ is defined as in Assumption A2. Suppose $y = x$. Then

$$\begin{aligned}\sum_{n=0}^{\infty} \tilde{b}(n) &= \sum_{n=0}^{\infty} e^{\theta n} b(n) \\ &= \sum_{n=0}^{\infty} e^{\theta n} P_x(X_n = x, T \wedge \tau > n) \\ &= E_x \sum_{n=0}^{T\wedge\tau-1} I(X_n = x) e^{\theta n} \\ &= 1,\end{aligned}$$

since $X_n \neq x$ for $1 \leq n < \tau$. Under Assumptions A2 and A3, the renewal theorem therefore applies (see, for example, Feller (1968)), permitting us to conclude that if $y = x$, then

$$\tilde{a}(n) \to \sum_{j=0}^{\infty} \tilde{b}(j) / \sum_{j=0}^{\infty} j\tilde{g}(j)$$

as $n \to \infty$. We have therefore established the following result.

Proposition 1 *Under Assumptions A2 and A3,*

$$P_x(X_n = x, T > n) \sim e^{-\theta n} / E_x[\tau e^{\theta\tau}; T > \tau]$$

as $n \to \infty$.

Comparing with Equation 2, we conclude that $\lambda = \theta$. Consequently, the large deviations exponent λ may be characterized as the root λ of the equation

$$E_x[e^{\lambda\tau}; T > \tau] = 1$$

This characterization will form the basis of our numerical approach to computing λ.

Before proceeding to a discussion of the algorithm, we wish to argue that under Assumption A1, if Assumptions A2 and A3 holds for one particular "regeneration state" x, then Assumptions A2 and A3 holds for all $x \in A^c$. Without such a guarantee, the choice of regeneration state would be complicated greatly.

Observe that if Assumption A2 holds for some $x \in A^c$, then for $y \in A^c$,

$$\begin{aligned}& E_y[e^{\theta\tau}; T > \tau] \\ = & \sum_{z\neq x, z\in A^c} e^{\theta} P(y,z) E_z[e^{\theta\tau}; T > \tau] + e^{\theta} P(y,x) \\ = & e^{\theta} \sum_{z\in A^c} P(y,z) E_z[e^{\theta\tau}; T > \tau]\end{aligned}$$

Hence, if we set $h(y) = E_y[e^{\theta\tau}; T > \tau]$, we may conclude that for $y \in A^c$,

$$\sum_{z\in A^c} P(y,z) h(z) = e^{-\theta} h(y).$$

In other words, h is a (column) eigenvector of B corresponding to eigenvalue $e^{-\theta}$. Furthermore, under Assumption A1, it is trivially evident that h is strictly positive. Also, h is finite-valued under Assumption A1, because

$$\begin{aligned} 1 = h(x) &\geq E_x[e^{\theta\tau}; \tau_y < \tau < T] \\ &\geq P_x(\tau_y < \tau < T)h(y) \end{aligned}$$

where $\tau_y = \inf\{n \geq 1 : X_n = y\}$. Then we have established the following result.

Proposition 2 *Assume that Assumption A2 holds, for some $x \in A^c$. Then under Assumption A1, there exists a strictly positive finite-valued (column) eigenvector of B such that*

$$Bh = e^{-\theta}h.$$

Proposition 2 establishes that the large deviations exponent can be characterized in terms of the Perron-Frobenius eigenvalue of B. Thus, one potential means of computing the large deviations exponent would involve computing the solution of the above eigenvalue problem.

Let $R = (R(y,z) : y, z \in A^c)$ be the matrix with entries

$$R(y,z) = e^{\theta}B(y,z)h(z)/h(y).$$

It is easily verified that R is irreducible and stochastic, under our assumptions. Let $\tilde{P}_y(\cdot)$ and $\tilde{E}_y(\cdot)$ denote the probability distribution and expectation on the path-space of X under which $X_0 = y$ and X evolves according to one-step transition matrix R. Then, for $y \in A^c$, it is evident that

$$\tilde{E}_y Z = E_y[Z\frac{h(X_n)}{h(X_0)}; T > \tau] \cdot e^{\theta n}$$

for any non-negative $\mathcal{F}_n$-measurable r.v. Z ($\mathcal{F}_n \stackrel{\Delta}{=} \sigma(X_0, \cdots, X_n)$). The above relation can be easily extended to $\mathcal{F}_{\tau_z}$-measurable Z:

$$\tilde{E}_y[Z; \tau_z < \infty] = E_y[Ze^{\theta\tau_z}\frac{h(X_{\tau_z})}{h(X_0)}; T > \tau_z, \tau_z < \infty] = \frac{h(z)}{h(y)}E_y[Ze^{\theta\tau_z}; T > \tau_z]. \tag{6}$$

(We used here the fact that $T > \tau_z$ is incompatible with $\tau_z = +\infty$.) Setting $y = z = x$ and $z \equiv 1$, we arrive at the identity

$$\tilde{P}_x(\tau_x < \infty) = E_x[e^{\theta\tau}; T > \tau] = 1.$$

Consequently, Assumption A2 guarantees that R is a recurrent matrix. Under the irreducibility of Assumption A1, we know that recurrence is a "solidarity" property, so that $\tilde{P}_y(\tau_y < \infty) = 1$ for $y \in A^c$. Relation 6 therefore implies that

$$E_y[e^{\theta\tau_y}; T > \tau_y] = 1$$

for $y \in A^c$.

Setting $y = z = x$ and $Z = \tau$, equation 6 yields

$$\tilde{E}_x(\tau; \tau < \infty) = E_x[\tau e^{\theta\tau}; T > \tau] < \infty$$

Thus, if Assumption A2 and A3 hold for some $x \in A^c$, then R is positive recurrent. Since positive recurrence is also a "solidarity" property, evidently $\tilde{E}_y(\tau < \infty)$ for $y \in A^c$. Relation 6 therefore implies that

$$\infty > \tilde{E}_y \tau_y = E_y[\tau_y e^{\theta\tau_y}; T > \tau_y],$$

verifying Assumption A3 for $y \in A^c$. We have therefore proved the following result.

Proposition 3 *Assume that Assumption A1 holds. Then,*

1. *if Assumption A2 is valid for one $x \in A^c$, Assumption A2 is valid for all $x \in A^c$.*
2. *if Assumption A3 is valid for one $x \in A^c$, Assumption A3 is valid for all $x \in A^c$.*

Before concluding this section, we note that we have essentially established that under Assumption A1, A2 is equivalent to requiring that B is R-recurrent (with $R = e^{\theta}$), whereas Assumptions A2 and A3 together imply that B is R-positive (again, with $R = e^{\theta}$). The concepts of R-recurrence and R-positivity are fundamental to the theory of quasi-stationary distributions; see for example, Seneta (1980).

In the next section, we describe our simulation-based estimator for computing the large deviations exponent λ (or equivalently, θ).

4 A Simulation-Based Estimator

If B is irreducible, then Proposition 3 establishes that if

$$E_y[e^{\theta\tau_y}; \tau_y < T] = 1 \tag{7}$$

holds for one $y \in A^c$, then Equation 7 holds for all $y \in A^c$. So, fix $x \in A^c$, and consider simulating n i.i.d. replicates of X up to time $T \wedge \tau_x$, conditioned on starting in state x. In particular, let

$$W_i(\gamma) \stackrel{\triangle}{=} e^{\gamma\tau_i} I(\tau_i < T_i)$$

be the value of the r.v. $e^{\gamma\tau_x} I(\tau_x < T)$ associated with i-th replicate. Set

$$\bar{W}_i(\gamma) = \frac{1}{n} \sum_{i=1}^{n} W_i(\gamma).$$

Clearly, $\bar{W}_n(\cdot)$, is strictly increasing and continuous, with $\bar{W}_n(-\infty) = 0$ and $\bar{W}_n(\infty) = +\infty$. Hence, for each n, there exists a unique root λ_n to the equation

$$\bar{W}_n(\lambda_n) = 1$$

It is easy to show that the empirical root λ_n is consistent for λ.

Proposition 4 *Under Assumptions A1 and A2, $\lambda_n \to \lambda$ almost surely as $n \to \infty$, where λ solves Equation 7.*

Proof: For $\varepsilon > 0$, the strong law of large numbers guarantees that

$$\begin{aligned} \bar{W}_n(\lambda - \varepsilon) &\to E_x[e^{(\lambda-\varepsilon)\tau_x}; \tau_x < T] < 1 \\ \bar{W}_n(\lambda + \varepsilon) &\to E_x[e^{(\lambda+\varepsilon)\tau_x}; \tau_x < T] > 1 \end{aligned}$$

as $n \to \infty$. Consequently, $\lambda_n \in (\lambda - \varepsilon, \lambda + \varepsilon)$ for n sufficiently large. Since ε was arbitrary, this proves the result. □

We turn next to the central limit theorem (CLT) for λ_n. Assume that:

A4 $E_x[e^{2\lambda\tau_x}; \tau_x < \tau] < \infty$.

Theorem 2 *Under Assumptions A1, A2 and A4,*

$$n^{1/2}(\lambda_n - \lambda) \Rightarrow \frac{\sqrt{Var_x W_1(\lambda)}}{w'_x(\lambda)} N(0,1)$$

as $n \to \infty$, where $w'_x(\lambda) = E_x[\tau_x e^{\lambda\tau_x}; \tau_x < T]$ and $Var_x(\cdot) \triangleq Var(\cdot | X_0 = x)$.

Proof: Note that

$$\bar{W}_n(\lambda_n) - \bar{W}_n(\lambda) = 1 - \bar{W}_n(\lambda),$$

and consequently

$$\bar{W}'_n(\xi_n)(\lambda_n - \lambda) = 1 - \bar{W}_n(\lambda), \tag{8}$$

where ξ_n lies between λ and λ_n. Now,

$$\bar{W}'_n(\xi_n) = \frac{1}{n} \sum_{i=1}^{n} \tau_i e^{\xi_n \tau_i} I(\tau_i < T_i).$$

Because $\bar{W}'_n(\cdot)$ is strictly increasing and $\lambda_n \to \lambda$ a.s. as $n \to \infty$, it is evident that for $\varepsilon > 0$ and n large enough,

$$\bar{W}'_n(\lambda - \varepsilon) \leq \bar{W}'_n(\xi_n) \leq \bar{W}'_n(\lambda + \varepsilon).$$

But the strong law of large numbers ensures that

$$\begin{aligned} \bar{W}'_n(\lambda - \varepsilon) &\to w'_x(\lambda - \varepsilon), \\ \bar{W}'_n(\lambda + \varepsilon) &\to w'_x(\lambda + \varepsilon) \end{aligned}$$

as $n \to \infty$. Since ε was arbitrary, we may conclude that $\bar{W}_n(\xi_n) \to w'_x(\lambda)$ a.s. as $n \to \infty$. On the other hand, Assumption A4 ensures that

$$n^{1/2}(1 - \bar{W}_n(\lambda)) \Rightarrow \sqrt{Var_x W_1(\lambda)} N(0,1) \tag{9}$$

as $n \to \infty$. Combining Equations 8 and 9 yields the desired conclusion. □

Given Proposition 3, it is perhaps surprising that the "second moment" hypothesis Assumption A4 is not a solidarity property in the presence of Assumption A1. In particular, it is easy to construct examples in which Assumption A4 holds for one $x \in A^c$ but not all $x \in A^c$, even in the presence of irreducibility. This raises the question of what happens when Assumption A4 fails to hold. We will study this issue further in future research.

Given the CLT provided by Theorem 2, we may construct confidence intervals for the large deviations exponent λ. In particular, let

$$V_n(\gamma) = \frac{1}{n-1} \sum_{i=1}^{n} (W_i(2\gamma) - \bar{W}_n(\gamma))^2.$$

It is easily verified that under the conditions of Theorem 2 (plus the additional proviso that $Var_x W_1(\lambda) > 0$),

$$[\lambda_n - z\frac{\sqrt{V_n(\lambda_n)}}{\bar{W}_n'(\lambda_n)}, \lambda_n + z\frac{\sqrt{V_n(\lambda_n)}}{\bar{W}_n'(\lambda_n)}]$$

is an asymptotic (as $n \to \infty$) $100(1-\delta)\%$ confidence interval for λ, where z solves the equation $P(N(0,1) \leq z) = 1 - \delta/2$.

References

[1] Feller, W. (1968). "An Introduction to Probability Theory and its Applications, Volumn 1" (3rd edition). John Wiley, New York.

[2] Kingman, J.F.C. (1963). The exponential decay of Markov transition probabilities. Proc. London Math. Soc. (3) 13, 337-338.

[3] Ney, P. and E. Nummelin (1987). Markov additive processes I: Eigenvalue properties and limit theorems. Annals of Probability 15, 561-592.

[4] Seneta, E. (1980) Non-negative Matrices and Markov Chains. Springer-Verlag, New York.

[5] Vere-Jones, D. (1962). Geometric ergodicity in denumerable Markov chains, Quart. J. Math. Oxford (2nd sec.) 13, 7-28.

Extensions of GSMP Models for Broadband Network Traffic Modeling

Charles A. Brooks

Department of Electrical Engineering and Computer Science
University of Illinois at Chicago
cbrooks@eecs.uic.edu
http://www.eecs.uic.edu/~cbrooks

October 2, 1996

Abstract

We introduce and compare relevant aspects of two new classes of Generalized Semi-Markov Processes (GSMPs). These new classes are extensions, adaptations, and generalizations of the classical GSMP model put forth by Matthes, Whitt, and others. We refer to these new classes as Doubly-Stochastic GSMPs (DS-GSMPs) and Doubly-Stochastic GSMPs with Filtering. These processes can be used as parsimoniously parameterized models for processes with short-range dependent or long-range dependent characteristics. Long-range dependent characteristics have been observed in a number of traces generated by traffic measurements on a variety of computer communication networks, and are relevant when considering models for broadband (particularly ATM) network traffic.

1 Introduction

Generalized semi-Markov processes have been of interest to the research community dating back to their introduction by Matthes [12] in the early sixties. Since then their use as stochastic models for complex systems has been both solidified (e.g., see Whitt [16] and the references therein) and demonstrated (e.g., see [8] and the references therein). A long-standing purpose of GSMPs has been to function as a model for discrete-event simulation [9, 2].

Our aim in this paper is to exhibit the suitability of GSMPs for modeling the external traffic arrivals to broadband networks, which are a class of computer communication networks that are characterized by their ability to provide an integrated set of services having wide-ranging performance requirements [3]. To this end, we present several traffic models specified as GSMPs. We illustrate models that are representative of both short-range dependent (SRD) and long-range dependent (LRD) traffic.

We show by example that traditional GSMPs can be used to represent a diverse set of SRD traffic models. In addition, to provide parsimoniously parameterized specifications of LRD traffic models, we introduce Doubly-Stochastic GSMPs (DS-GSMPs), which by construction may easily

incorporate additional functions such as filtering.

2 Short-Range and Long-Range Dependence

Our aim is to provide an extension to the traditional GSMP model that will allow it to represent both short-range dependent and long-range dependent traffic models, and to do so without an excessive number of parameters. Keeping this goal in mind, let us provide some background information. Let $X = \{X_k : k = 1, 2, \ldots\}$ be a covariance stationary stochastic process with mean m, variance σ^2, and autocorrelation function $r(k)$. For our purposes, X_k represents the number of arrivals during the kth (fixed) time interval of length T. I.e., X_k is the number of arrivals during time interval $\{t : (k-1)T \leq t < kT\}$.

Until recent years traffic models for computer communication networks were almost always short-range dependent. SRD models such as Markovian processes, are characterized by autocorrelation functions that decay exponentially fast (i.e., $r(k) \sim \exp(k)$). This implies that the spectral density function is bounded at the origin. For a comprehensive survey of SRD models, see [6].

Within the last few years, LRD models have been proposed by many researchers to be better suited for network trafffic models. This proposition is based on various traffic observations, including observations of traffic on telecommunication signaling networks [4], Ethernet local area network measurements [10], traces from wide area networks [13], and variable-bit-rate (VBR) video [1]. LRD models, such as self-similar models, are characterized by autocorrelation functions that decay hyperbolically (i.e., $r(k) \sim k^{-D}$, where $0 < D < 1$). This implies that the spectral density function obeys a power law and is unbounded at the origin. Recently, Erramilli, Narayan, and Willinger [5] have shown conditions in which parsimonious LRD modeling of network traffic is justified.

3 Traditional GSMPs

Prior to presenting examples of GSMP-based traffic models, let us first introduce our notation by reviewing the traditional GSMP (with Speeds) [16, 15]. A Generalized Semi-Markov Process with Speeds is characterized by the following septuple,

$$(S, A, \xi(\cdot), p(\cdot;\cdot,\cdot), p_0(\cdot), F_{(\cdot)}(\cdot), r_{(\cdot)}(\cdot)),$$

where:

S is a state space (finite or countable) representing the set of physical states of a system.

A is a finite set of events.

$\xi(s)$ is the set of possible events in state s ($\xi(s) \subseteq A$).

$p(s'; s, \alpha')$ is the probability of jumping to state s' when event α' occurs in state s.

$p_0(s)$ is the initial state distribution.

$\{F_{\alpha'}(t), \alpha' \in A\}$ is the set of distributions (one for each $\alpha' \in A$) on the non-negative real line used to generate new clock samples for events.

$\{r_{\alpha'}(s), \alpha' \in A\}$ is the set of speed functions (one for each $\alpha' \in A$).

The evolution of the GSMP is divided into stages:

1. Determine the initial state, s_0, using distribution $p_0(s)$;

2. Determine the feasible event set, $\xi(s_0)$;

3. For all feasible events, $\alpha' \in \xi(s_0)$, generate event clock values based on the respective clock distributions; (All remaining event clocks are set to zero;) The current state, s, is assigned the value of s_0.

4. Determine the triggering event α'' by finding the feasible event with smallest triggering time, where the triggering time for an event is defined as the event's clock value divided by its clock speed in state s;

5. Determine the next state of the process, s', by using the probability mass function $p(s'; s, \alpha'')$;

6. Determine the new feasible event set, $\xi(s')$;

7. Update the event clock values; (New clock values are generated for all "new" events, which are the elements of the set $\xi(s') \setminus (\xi(s) - \{\alpha'\})$, get new clock values according to their respective distributions, while all "old" events, which are the elements of the set $\xi(s') \cap (\xi(s) - \{\alpha'\})$, keep their current clock values. The remaining event clocks are set to zero.) The current state, s, is assigned the value of s'.

8. Return to the step (3) and continue.

This cycle is executed repeatedly. Under suitable conditions this process evolves over an infinite time horizon.

4 Models with Short-Range Dependence

Here are simple GSMP specifications for three typical SRD traffic models:

Example 4.1 *Poisson Process*

Assume that the process has a rate of λ.

- $S = \{s_0\}$; There is a single state, the initial state;
- $A = \{\alpha\}$; There is a single event, the arrival event;
- $\xi(s) = A, \forall s \in S$;
- $p(s_0; s, \alpha') = 1, \forall s \in S, \alpha' \in A$;
- $p_0(s_0) = 1$;
- $F_\alpha(t) = [1 - \exp(-\lambda t)] \cdot u(t)$, where $u(\cdot)$ denotes the unit step function.
- $r_\alpha(s) = 1, \forall s \in S$;

Example 4.2 *Markov-Modulated Fluid Process*

Assume that the Markov chain has N states, $s_1, s_2, \ldots, s_N$, and V is the set of Markov chain transitions.

- $S = \{s_1, s_2, \ldots, s_N\}$;
- $A = \{\beta_{\{j,k\}} : j \to k \in V\}$;
- $\forall j \in \{1, \ldots, N\}$, $\xi(s_j) = \{k : j \to k \in V\}$;
- $\forall \{j, k\} \in \{1, \ldots, N\} \times \{1, \ldots, N\}$, $p(k; j, \beta_{\{j,k\}}) = 1$, if $j \to k \in V$, and $p(k; j, \beta_{\{j,k\}}) = 0$, otherwise;
- $p_0(s_1) = 1$;
- $F_{\beta_{\{j,k\}}}(t) = [1 - \exp(-\lambda_{\{j,k\}} t)] \cdot u(t)$;
- $r_{\beta_{\{j,k\}}}(s) = 1, \forall s \in S$;

Note that for Markov chains, we can further reduce the number of parameters by uniformization.

Example 4.3 *On-Off Markov-Modulated Poisson Process*

Assume that the Markov chain has 2 states, off and on. In the off state, the process is idle. In the on state, the process is a Poisson process with rate λ.

- $S = \{s_{off}, s_{on}\}$;
- $A = \{\alpha, \beta_{\{off \to on\}}, \beta_{\{on \to off\}}\}$, where α represents an arrival, and the two remaining events represent transitions in the Markov chain state;
- $\xi(s_{off}) = \{\beta_{\{off \to on\}}\}$, and $\xi(s_{on}) = \{\alpha, \beta_{\{on \to off\}}\}$;
- $p(s_{on}; s_{on}, \alpha) = 1$, $p(s_{on}; s_{off}, \beta_{\{off \to on\}}) = 1$, $p(s_{off}; s_{on}, \beta_{\{on \to off\}}) = 1$, and $p(\cdot; \cdot, \cdot) = 0$, otherwise;
- $p_0(s_{off}) = 1$;
- $F_\alpha = [1 - \exp(-\lambda t)] \cdot u(t)$, $F_{\beta_{\{off \to on\}}}(t) = [1 - \exp(-\lambda_{\{off \to on\}} t)] \cdot u(t)$, and $F_{\beta_{\{on \to off\}}}(t) = [1 - \exp(-\lambda_{\{on \to off\}})] \cdot u(t)$;
- $r_{\alpha'}(s) = 1, \forall \{s, \alpha'\} \in S \times A$;

4.1 Superposition of GSMP State

Typically network traffic is the aggregation of several traffic streams. We can easily incorporate this in the GSMP model (as shown in the following example):

Example 4.4 *Poisson Process Revisited*

Assume that there are M processes, and process j has a corresponding rate of λ_j, where $j \in \{1, \ldots, M\}$;

- $S = \{s_0\}$;
- $A = \{\alpha_j : j \in \{1, \ldots, M\}\}$, where α_j corresponds to the arrival event in process j;
- $\xi(s) = A, \forall s \in S$;
- $p(s_0; s, \alpha') = 1, \forall s \in S, \alpha' \in A$;
- $p_0(s_0) = 1$;
- $F_{\alpha_j}(t) = [1 - \exp(-\lambda_j t)] \cdot u(t)$, $\forall j \in \{1, \ldots, M\}$;
- $r_{\alpha_j}(s) = 1, \forall \{s, j\} \in S \times \{1, \ldots, M\}$;

Note that for aggregate Poisson processes, we can further reduce the number of parameters by uniformization. However, this reduction is not possible for processes that are not Markov.

5 Models with Long-Range Dependence

Here are simple specifications for three LRD processes [14, 11]:

Example 5.1 *Fractal Renewal Process [14]*

This process is a renewal process with epochs determined by a heavy-tailed distribution, which in this case is the Pareto distribution, with parameters γ and δ.

- $S = \{s_0\}$;
- $A = \{\alpha\}$;
- $\xi(s) = A, \forall s \in S$;
- $p(s_0; s, \alpha') = 1, \forall s \in S, \alpha' \in A$;
- $p_0(s_0) = 1$;
- $F_\alpha(t) = \left[1 - \left(\frac{\gamma}{t}\right)^{-\delta}\right] \cdot u(t - \gamma)$;
- $r_\alpha(s) = 1, \forall s \in S$;

Example 5.2 *Pareto-Modulated Deterministic Process [11]*

This process has no arrivals in its idle state, and has deterministic interarrival times otherwise.

- $S = \{s_{off}, s_{on}\}$;
- $A = \{\alpha, \beta_{\{off \to on\}}, \beta_{\{on \to off\}}\}$;
- $\xi(s_{off}) = \{\beta_{\{off \to on\}}\}$, and
  $\xi(s_{on}) = \{\alpha, \beta_{\{on \to off\}}\}$;
- $p(s_{on}; s_{on}, \alpha) = 1$,
  $p(s_{on}; s_{off}, \beta_{\{off \to on\}}) = 1$,
  $p(s_{off}; s_{on}, \beta_{\{on \to off\}}) = 1$, and
  $p(\cdot; \cdot, \cdot) = 0$, otherwise;
- $p_0(s_{off}) = 1$;
- $F_\alpha(t) = u(t-1)$,
  $F_{\beta_{\{off \to on\}}}(t) =$
  $\left[1 - \left(\frac{\gamma_{\{off \to on\}}}{t}\right)^{-\delta_{\{off \to on\}}}\right] \cdot$
  $u(t - \gamma_{\{off \to on\}})$, and
  $F_{\beta_{\{on \to off\}}}(t) =$
  $\left[1 - \left(\frac{\gamma_{\{on \to off\}}}{t}\right)^{-\delta_{\{on \to off\}}}\right] \cdot$
  $u(t - \gamma_{\{on \to off\}})$;
- $r_\alpha(s_{on})$ equals the packet arrival rate during the *on* period, and $r_{\alpha'}(s) = 1$, otherwise;

Example 5.3 *Fractal Binomial-Noise-Driven Poisson Process [14]*

This is an example of a Pareto-modulated Poisson Process.

- $S = \{s_{off}, s_{on}\}$;
- $A = \{\alpha, \beta_{\{off \to on\}}, \beta_{\{on \to off\}}\}$, where α represents an arrival, and the two remaining events represent transitions in the modulation chain state;
- $\xi(s_{off}) = \{\beta_{\{off \to on\}}\}$, and
  $\xi(s_{on}) = \{\alpha, \beta_{\{on \to off\}}\}$;
- $p(s_{on}; s_{on}, \alpha) = 1$,
  $p(s_{on}; s_{off}, \beta_{\{off \to on\}}) = 1$,
  $p(s_{off}; s_{on}, \beta_{\{on \to off\}}) = 1$, and
  $p(\cdot; \cdot, \cdot) = 0$, otherwise;
- $p_0(s_{off}) = 1$;
- $F_\alpha(t) = 1 - \exp(-\lambda t) \cdot u(t)$,
  $F_{\beta_{\{off \to on\}}}(t) =$
  $\left[1 - \left(\frac{\gamma_{\{off \to on\}}}{t}\right)^{-\delta_{\{off \to on\}}}\right] \cdot$
  $u(t - \gamma_{\{off \to on\}})$, and
  $F_{\beta_{\{on \to off\}}}(t) =$
  $\left[1 - \left(\frac{\gamma_{\{on \to off\}}}{t}\right)^{-\delta_{\{on \to off\}}}\right] \cdot$
  $u(t - \gamma_{\{on \to off\}})$;
- $r_{\alpha'}(s) = 1, \forall \{s, \alpha'\} \in S \times A$;

5.1 Doubly-Stochastic GSMPs

Now consider what we have termed a Doubly-Stochastic GSMP (DS-GSMP). It is a GSMP in which the clock distributions for each event have two parameters, t and g, where $g = g(\cdot)$ is a function that maps S onto the real line. Note that the image of g is countable because we have assumed that S is countable. A Doubly-Stochastic GSMP is characterized by the following octuple,

$$(S, A, \xi(\cdot), p(\cdot;\cdot,\cdot), p_0(\cdot), g(\cdot),$$
$$F_{(\cdot)}(\cdot,\cdot), r_{(\cdot)}(\cdot)),$$

where:

S is a state space.

A is a finite set of events.

$\xi(s)$ is the set of possible events function.

$p(s'; s, \alpha')$ is the probability mass function.

$p_0(s)$ is the initial state distribution.

$g(s)$ is the rate function.

$\{F_{\alpha'}(t, g), \alpha' \in A\}$ is the set of event clock distributions. (Assume that $F_{\alpha'}(t, \cdot)$ is a well-defined distribution for all values of $g \in (a, b)$, where $0 < a < b \leq \infty$.)

$\{r_{\alpha'}(s), \alpha' \in A\}$ is the set of speed functions.

If the probability mass function for a DS-GSMP defines a deterministic random variable, then it is possible to derive both an analogous discrete-time Markov process and its corresponding continuous-time Markov process [16]. It is also true that the set of all stochastic processes that can be expressed by a DS-GSMP is equal to the set of all stochastic processes that can be expressed by a GSMP. It is easy to see how the DS-GSMP below may be written as a traditional GSMP by representing each possible state transition as an event.

Example 5.4 *GSMP-Modulated Poisson Process*

Let us consider the superposition of M Fractal Binomial-Noise-Driven Poisson Processes. Define $I(s) \subseteq \{1, \ldots, M\}$, as to represent the set of sources that are idle.

- $S = \{s^1_{off}, s^1_{on}\} \times \{s^2_{off}, s^2_{on}\} \times \ldots \times \{s^M_{off}, s^M_{on}\}$;
- $A = \{\alpha, \beta^1_{\{off \to on\}}, \beta^2_{\{off \to on\}}, \ldots, \beta^M_{\{off \to on\}}, \beta^1_{\{on \to off\}}, \beta^2_{\{on \to off\}}, \ldots, \beta^M_{\{on \to off\}}\}$;
- $\alpha \in \xi(s)$, $\forall s \in \{s : I(s) \neq \{1, 2, \ldots, M\}\}$;
  $\forall j \in \{1, 2, \ldots, M\}, \beta^j_{\{off \to on\}} \in \xi(s)$, $\forall s \in \{s : j \in I(s)\}$;
  $\forall j \in \{1, 2, \ldots, M\}, \beta^j_{\{on \to off\}} \in \xi(s)$, $\forall s \in \{s : j \notin I(s)\}$;
- $p(s; s, \alpha) = 1$, $\forall s \in \{s : I(s) \neq \{1, 2, \ldots, M\}\}$; $\forall j \in \{1, 2, \ldots, M\}$, $p(\{s^1, s^2, \ldots, s^j_{on}, \ldots, s^M\}; \{s^1, s^2, \ldots, s^j_{off}, \ldots, s^M\}, \beta^j_{\{off \to on\}}) = 1$, where, $s^i \in \{s^i_{off}, s^i_{on}\}$; $\forall j \in \{1, 2, \ldots, M\}$, $p(\{s^1, s^2, \ldots, s^j_{off}, \ldots, s^M\}; \{s^1, s^2, \ldots, s^j_{on}, \ldots, s^M\}, \beta^j_{\{on \to off\}}) = 1$, where, $s^i \in \{s^i_{off}, s^i_{on}\}$; and $p(\cdot; \cdot, \cdot) = 0$, otherwise;
- $p_0(\{s^1_{off}, s^2_{off}, \ldots, s^M_{off}\}) = 1$;
- $g(s) = \lambda \cdot [M - \Gamma(I(s))]$, where $\Gamma(\cdot)$ is the cardinality function;
- $F_\alpha(t, g) = [1 - \exp(-gt)] \cdot u(t)$, $\forall j \in \{1, 2, \ldots, M\}$, $F_{\beta^j_{\{off \to on\}}}(t, g) = \left[1 - \left(\frac{\gamma^j_{\{off \to on\}}}{t}\right)^{-\delta^j_{\{off \to on\}}}\right] \cdot u(t - \gamma^j_{\{off \to on\}})$, and $\forall j \in \{1, 2, \ldots, M\}$, $F_{\beta^j_{\{on \to off\}}}(t, g) = \left[1 - \left(\frac{\gamma^j_{\{on \to off\}}}{t}\right)^{-\delta^j_{\{on \to off\}}}\right] \cdot u(t - \gamma^j_{\{on \to off\}})$;
- $r_{\alpha'}(s) = 1, \forall \{s, \alpha'\} \in S \times A$;

Note the reduction in the number of events from $O(M^3)$ to $O(M)$.

5.2 DS-GSMPs with Filtering

The DS-GSMP model is no more expressive than that of the GSMP, but it does facilitate the introduction of non-Markov rate functions. Excusing the abuse of notation, let $\{s_{t'}, t' > 0\}$, be the evolution of the state trajectory, and let $\{g_{t'}, t' > 0\}$, be the evolution of the rate function. Now consider a family of non-causal filter functions $\{h_t : t > 0\}$. For any $t > 0$, h_t is defined as a function whose domain is the set of all possible rate function trajectories of $g_{t'}$ over the finite horizon $[0, t]$ and whose range is the set of non-negative real numbers. If we define the clock event distributions to be a function of t and h_t, this leads to the following definition:

A DS-GSMP with Filtering is characterized by the following nontuple,

$$(S, A, \xi(\cdot), p(\cdot;\cdot,\cdot), p_0(\cdot), g(\cdot),$$
$$h_{(\cdot)}(\cdot), F_{(\cdot)}(\cdot,\cdot), r_{(\cdot)}(\cdot)),$$

where:

S is a state space.

A is a finite set of events.

$\xi(s)$ is the set of possible events function.

$p(s'; s, \alpha')$ is the probability mass function.

$p_0(s)$ is the initial state distribution.

$g(s)$ is the rate function.

$h(\{g_{t'}, t' > 0\})$ is a filter.

$\{F_{\alpha'}(t, h), \alpha' \in A\}$ is the set of event clock distributions. (Assume that $F_{\alpha'}(t, \cdot)$ is a well-defined distribution for all values of $h \in (a, b)$, where $0 < a < b \leq \infty$.)

$\{r_{\alpha'}(s), \alpha' \in A\}$ is the set of speed functions.

If $\{h_t : t > 0\}$ is chosen appropriately, the DS-GSMP may exhibit long-term memory and as a result the model may have long-range dependence. Note that in this case the process can not be regenerative [15, 7]. Consider the following example that is based on a Fractal Shot-Noise-Driven Poisson Process [14].

Example 5.5 *Fractal Filtered-Rate-Driven Poisson Process*

- $S = \{s_0, s_1, s_2, \ldots\}$;
- $A = \{\alpha\}$;
- $\xi(s) = A, \forall s \in S$;
- $p(s_{i+1}; s_i, \alpha') = 1, \forall i \geq 0, \alpha' \in A$;
- $p_0(s_0) = 1$;
- $g(s_i) = i$;
- $h_t(\{g_{t'} : 0 \leq t' \leq t\}) =$
$$\int_0^t \left[\left(\frac{d}{dt'} g(t') \right) \cdot (t - t')^{-\left(1 - \frac{\delta}{2}\right)} \right] dt'$$
- $F_\alpha(t, h) = [1 - \exp(-ht)] \cdot u(t)$, $\forall j \in \{1, 2, \ldots, M\}$,
- $r_{\alpha'}(s) = 1, \forall \{s, \alpha'\} \in S \times A$;

6 Conclusions

This paper introduces two new classes of GSMPs, termed DS-GSMPs and DS-GSMPs with filtering. These extensions of traditional GSMPs allow frugal descriptions of processes that normally require an excessively paramterized GSMP specification or processes that can not be modeled at all by a (finite) GSMP specification. By adapting GSMPs and extending their expressiveness, there exists a common framework for the analysis of LRD processes by discrete event simulation (DES) and existing theoretical results for non-regenerative stochastic processes.

References

[1] J. Beran, R. Sherman, M. S. Taqqu, and W. Willinger. Long-range dependence in variable-bit-rate video traffic. *IEEE Trans. Comm.*, 43(2/3/4):1566–1579, February/March/April 1995.

[2] C. A. Brooks, S. Wei, and H. Li. AND - ATM Network Design System. Technical Memorandum UIC-EECS-96-2, University of Illinois at Chicago, Department of Electrical Engineering and Computer Science, May 1996. Submitted to Simulation Theory and Practice.

[3] M. de Prycker. *Asynchronous Transfer Mode: Solution for Broadband ISDN.* Prentice Hall, 1995.

[4] D. Duffy, A. McIntosh, M. Rosenstein, and W. Willinger. Statistical analysis of CCSN/SS7 traffic data from working CCS subnetworks. *IEEE J. Sel. Areas Comm.*, 12(3):544–551, April 1994.

[5] A. Erramilli, O. Narayan, and W. Willinger. Experimental queueing analysis with long-range dependent packet traffic. *IEEE/ACM Trans. Networking*, 4(2):209–233, April 1996.

[6] V. S. Frost and B. Melamed. Traffic modeling for telecommunication networks. *IEEE Communications Magazine*, 32(3):70–81, March 1994.

[7] P. Glasserman. *Gradient Estimation Via Perturbation Analysis.* Kluwer Academic Publishers, 1991.

[8] P. W. Glynn. Special issue. *Discrete Event Dynamic Sys.: Th. and Appl. (J. Discrete Event Dynamic Systems)*, 6, 1996.

[9] D. L. Iglehart and G. S. Shedler. Simulation of non-Markovian systems. *IBM J. Res. Develop.*, 27(5):472–480, September 1983.

[10] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Trans. Networking*, 2(1):1–15, February 1994.

[11] N. Likhanov, B. Tsybakov, and N. D. Georganas. Analysis of an ATM buffer with self-similar ("fractal") input traffic. In *Proc. of IEEE Infocom*, pages 985–992, April 1995.

[12] K. Matthes. Zur Theorie der Bedienungsprozesse. *Trans. 3rd Prague Conf. on Inf. Thy. Stat. Dec. Fct.*, 1962. Prague.

[13] V. Paxson and S. Floyd. Wide area traffic: The failure of Poisson modeling. *IEEE/ACM Trans. Networking*, 3(3):226–244, June 1995.

[14] B. K. Ryu and S. B. Lowen. Point process approaches to the modeling and analysis of self-similar traffic - part I: Model construction. In *Proc. of IEEE Infocom*, pages 1468–1475, March 1996.

[15] G. S. Shedler. *Regenerative Stochastic Simulation.* Academic Press, 1993.

[16] W. Whitt. Continuity of generalized semi-Markov processes. *Math. Oper. Res.*, 5(4):494–501, 1980.

Synthesizing Control Code for Achieving State Targets in Composed Systems

Lawrence E. Holloway
Sriram Rajaram
Dept. of Electrical Engineering and Center for Manufacturing Systems,
University of Kentucky, Lexington, Kentucky 40506, USA

Abstract

This paper considers the problem of control synthesis for a class of discrete event systems composed of subsystems interacting through *condition* signals. The control objective is specified as a sequence of condition sets, and the purpose of control synthesis is to determine input signals to drive the composed system through a series of states that satisfies the given specification. The synthesis method developed builds up the control law by considering the system in layers. At each layer, a more detailed specification of the control objective is formulated, until the final result is a specification that fully defines the control actions for the system. The resulting specification is then automatically converted into a program for Programmable Logic Controllers.

1. Introduction and Motivation

Programmable Logic Controllers (PLCs) are commonly used for control of industrial equipment. These controllers primarily operate by sensing discrete (binary) signals from the process, and generating discrete (binary) actuation signals in response.

This paper describes continuing work in methods to automatically synthesize control code for low-level manufacturing systems such as would be controlled by a PLC. Given a model of the system and a goal of a sequence of conditions on the state that the system should achieve, our objective is to automatically determine a control law (PLC program) that will drive the system through these conditions. Such an approach allows frequent changes in control goals to be quickly and automatically implemented as control code.

The models we consider, element nets, are a type of *condition event* system, a form of Petri nets [1]. The system is represented as a set of individual nets. Each net inputs a set of condition signals which may disable or enable transition firings. Each net outputs condition signals which may be used as inputs to other element nets. The controller interacts with the systems in the same manner: sensing signals are conditions that are output by the nets, and control signals are conditions that are input to the nets. The modular specification of the models simplifies the modelling task by allowing reuse of common model components, and simplifies updating of models when subsystems are modified.

The control objective specifications we consider are given as sequences of condition sets that the system should achieve. The control approach developed in this paper is to

[0]Please address all correspondence to L. E. Holloway at the above address or email: holloway@engr.uky.edu. This work has been supported in part by Rockwell International, NSF grant ECS-9308737, and the Center for Manufacturing Systems at the University of Kentucky.

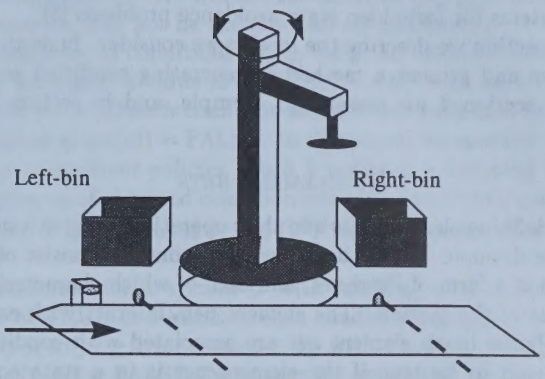


Fig. 1. A simple robot workcell example.

consider specification on a single net or subsystem. The nets are considered from the bottom up, i.e. we consider nets which are not inputs to other nets, and then we consider nets which generate input conditions to those nets, and then nets which input conditions to those nets, and so on. With each net, we determine the condition signal inputs to that net that must be used to achieve the control specification, and incorporate that into a specification used at the next highest net. After all nets have been considered, the specification includes condition and sequencing information necessary to define a control law.

To illustrate the type of systems we examine, consider the simple workcell example in figure 1. The example consists of a robot which moves parts from either of two bins to a product on a conveyor. Different types of products may arrive on the conveyor; some will require a part from the left bin of parts, and some will require a part from the right bin of parts.

The state of the workcell can be represented by the state of the robot arm position, the robot arm motor, the state of the two bins, the state of the gripper on the robot, the conveyor state, the state of the product on the conveyor, the state of the sensors, etc. Each of these elemental subsystems has its own set of states, and any state changes for such a subsystem may depend on the signals from the controller or conditions on the state of other subsystems. For example, the robot motor has states ML (motor on left), MR (motor on right), and MOFF (motor off). The state changes of the motor depend on signals coming from the controller. The robot arm position has the states LBin (over the left-bin), Home (over the conveyor), and RBin (over the right-bin), as well as intermediate states between these. Note that state changes in the arm position depend on the state of the robot motor. Similarly, when the gripper grasps a part, the next state of the gripper, representing the type of part held, depends on which bin the arm was over when the gripper was activated.

The goal of automatic control synthesis for manufacturing systems has been considered by several authors. Brandin, and later Leduc and Wonham, developed PLC implementations of language-based supervisory control laws [2; 3]. Klein, Jonsson, and Bäckström present planning methods for generating sequential control programs. They use a form

of propositional calculus, SAS+, for determining actions necessary to achieve a goal, and determine a class of systems for which the planning algorithm is of polynomial complexity [4]. Rausch, Lüder and Hanisch present a method of synthesizing controllers for condition event systems for forbidden state avoidance problems [5].

In the following section we describe the models we consider. In section 3 we formulate the control problem and present a method of generating condition sequences to define the controller. In section 4 we present an example, and in section 5 we discuss the implementation.

2. Element nets

Before formally defining element nets and their operation, we give a summary of several key ideas which we develop. First, the systems we consider consist of a set of element nets, each of which is a form of Petri net, and each of which characterizes the operation of different portions of the system. The element nets interact with each other through *condition labels*. Places in an element net are associated with condition labels, and a condition label is said to be true if the element net is in a state corresponding to a place with that label. Other element nets use these labels as firing conditions on their transitions: each transition has a set of substate conditions, output from other nets or from an external control, each of which must be true for a period of time for the transition to be enabled to fire.

Let $\mathcal{L}$ be a set of *condition labels*, and let $\mathcal{C} = \mathcal{L} \times \{0,1\}$ be the set of all conditions. Formally, we define an element net as a tuple $G = (\mathcal{P}_G, \mathcal{T}_G, \mathcal{E}_G, g, \varphi, r, M_G)$, where $\mathcal{P}_G$ is a set of places, $\mathcal{T}_G$ is a set of transitions, and $\mathcal{E}_G \subseteq (\mathcal{P}_G \times \mathcal{T}_G) \cup (\mathcal{T}_G \times \mathcal{P}_G)$ is a set of arcs. The *labeling function* g associates sets of labels to places, such that for all $p \in \mathcal{P}_G$, $g(p) \subseteq \mathcal{L}$. The function $\varphi : \mathcal{T}_G \to 2^{\mathcal{C}}$, associates an *enabling condition set* to each transition. The *residence time function* r gives a half-closed interval subset of $[0, \infty)$ to each transition in the net. Specifically, for any transition t, $r(t) = [r_{min}(t), r_{max}(t))$, where $r_{min}(t)$ denotes the minimum of the interval, and $r_{max}(t)$ denotes the limit of the maximum of the interval.

A marking $m : \mathcal{P}_G \to \{0, 1, \ldots\}$ of an element net G is a function that associates a nonnegative integer to each place in G. We will say that a place p is marked if $m(p)$ is nonzero. M_G is a set of initial markings for the net G, and will be discussed more later.

Given a place p and the set of arcs $\mathcal{E}_G$, the set of transitions which are outputs to p is denoted $p^{(t)} \subseteq \mathcal{T}_G$, and the set of transitions which are inputs to p is denoted by ${}^{(t)}p \subseteq \mathcal{T}_G$. An element net is a *state graph* if each transition has exactly one input place and exactly one output place.

A *condition system* consists of a nonempty set $\mathcal{G} = \{G_1, G_2, \ldots G_n\}$ of element nets over a common set of labels $\mathcal{L}$. Let $\mathcal{P}_\mathcal{G}$, $\mathcal{T}_\mathcal{G}$, and $\mathcal{E}_\mathcal{G}$ be the union of places, transitions, and arcs, respectively, among all element nets in $\mathcal{G}$. The function g for the system is the obvious extension of the labeling function from the individual nets over the domain $\mathcal{P}_1 \cup \mathcal{P}_2 \cup \cdots \mathcal{P}_n$. The functions φ and r are the obvious extensions of the functions on the individual nets to the domain $\mathcal{T}_1 \cup \mathcal{T}_2 \cup \cdots \mathcal{T}_n$. The set $M_\mathcal{G}$ is the set of all markings defined over $\mathcal{P}_\mathcal{G}$, such that for any $m \in M_\mathcal{G}$, the projection of m onto element net places $\mathcal{P}_{G_1}$ is a subset of M_{G_1}. Given a system $\mathcal{G}$, we generalize the notation such that for any $P \subseteq \mathcal{P}_\mathcal{G}$, $g(P) = \cup_{p \in P} g(p)$, and for any $T \subseteq \mathcal{T}$, $\varphi(T) = \cup_{t \in T} \varphi(t)$.

As mentioned above, a condition $c = (\ell, b)$ is a pair, for $\ell \in \mathcal{L}$ and $b \in \{0,1\}$. Given c, define $not(c)$ such that $not((\ell,1)) = (\ell,0)$ and $not((\ell,0)) = (\ell,1)$. For net $\mathcal{G}$, define $C_{(out,\mathcal{G})} := g(P_\mathcal{G}) \times \{0,1\}$. Given a set of nets $\mathcal{G}$, an *assignable set* of conditions for $\mathcal{G}$ is a set C such that such that $\varphi(\mathcal{T}_\mathcal{G}) - C_{(out,\mathcal{G})}$ and $C_{(out,\mathcal{G})} \cap C = \emptyset$, and $c \in C$ implies

$not(c) \in C$. Note that an assignable set C cannot include any conditions determined in $\mathcal{G}$, but must include any other conditions used for transition enabling. There is always a unique minimal assignable set. We will use the notation $C_{(in,\mathcal{G})}$ to refer to any assignable set for $\mathcal{G}$ (which may or may not be the minimal assignable set).

Given an assignable set of conditions $C_{(in,\mathcal{G})}$ for $\mathcal{G}$, an *external condition assignment* u for $\mathcal{G}$ over $C_{(in,\mathcal{G})}$ associates a value in {TRUE, FALSE} with each condition in $C_{(in,\mathcal{G})}$. Let $\mathcal{U}_{\mathcal{G},C_{(in,\mathcal{G})}}$ be the set of all such condition assignments such that for any $u \in \mathcal{U}_{\mathcal{G},C_{(in,\mathcal{G})}}$, $u(c) = TRUE$ implies $u(not(c))$ = FALSE. In this paper, we consider (untimed) dynamic *feedback condition assignment policies.* Such a policy is a mapping $U_{\mathcal{G},C_{(in,\mathcal{G})}} : (2^{\mathcal{C}})^* \rightarrow \mathcal{U}_{\mathcal{G},C_{(in,\mathcal{G})}}$ from sequences of observed condition sets onto an external condition assignment. ($2^{\mathcal{C}}$ indicates the set of all subsets of $\mathcal{C}$, and $(2^{\mathcal{C}})^*$ denotes the set of of all finite sequences of subsets of $\mathcal{C}$.)

A *condition evaluation* for a system $\mathcal{G}$ with marking m is a function $v_(\mathcal{G}, m)$ that associates a value of TRUE or FALSE to each condition in $C_{(out,\mathcal{G})}$, where $v_{\mathcal{G},m}((\ell,1)) = TRUE$ if and only if $m(p) \geq 1$ for some $p \in P_{\mathcal{G}}$ with $\ell \in g(p)$, and $v_{\mathcal{G},m}((\ell,0))$ = TRUE if and only if $v_{\mathcal{G},m}((\ell,1))$ = FALSE. Note that for any net $\mathcal{G}$, $C_{(in,\mathcal{G})} \cap C_{(out,\mathcal{G})} = \emptyset$.

We can now define the dynamics of our system models. Given a system $\mathcal{G}$, a transition set $T \in \mathcal{T}_{\mathcal{G}}$ is *marking enabled* if

$$m(p) \geq | \, p^{(t)} \cap T \, | \quad \text{for all } p \in {}^{(p)}t, \; t \in T.$$

The set T is *condition enabled* under marking m and condition assignment $u_{\mathcal{G},C_{(in,\mathcal{G})}}$ if for all $t \in T$ and all $c \in \varphi(t)$, either $u_{\mathcal{G},C_{(in,\mathcal{G})}}(c) = TRUE$ for $c \in C_{(in,\mathcal{G})}$ or $v_{\mathcal{G},m}(c) = TRUE$ otherwise. Thus, all conditions for a transition must be true either due to the marking or the external assignment.

The state of the system at a given time τ is represented as a triple $(m_\tau, T_\tau, clock_\tau)$, where m_τ is a marking, $T_\tau \subseteq \mathcal{T}$ is a set of transitions said to *fire* at time τ, and *clock* : $\mathcal{T} \rightarrow [0,\infty) \cup \{-1\}$ defines the *enabling clock values* for each transition in the system, where $clock(t) \in [0, r_{max}(t))$ for a transition t that is marking enabled and condition enabled at time τ, and $clock(t) = -1$ otherwise.

A *trajectory* of the system is a time function ξ such that $\xi(\tau) = (m_\tau, T_\tau, clock_\tau)$. Given a trajectory ξ over time τ_0 to time τ, define $C(\xi,\tau) \in (2^{\mathcal{C}})^*$ such that:

1. $C(\xi, 0) = \{c \mid v_{\mathcal{G},m_0}(c) = \text{TRUE}\}$
2. $C(\xi,\tau) = \begin{cases} C(\xi,\tau^-) & \text{if } C^\tau_{xi} = C^{\tau-}_\xi \\ C(\xi,\tau^-)C^\tau_\xi & \text{if } C^\tau_\xi \neq C^{\tau-}_\xi \end{cases}$

where $C^\tau_\xi := \{c \mid v_{\mathcal{G},m_\tau}(c) = \text{TRUE or } U(C(\xi,\tau^-))(c) = TRUE\} \in \mathcal{C}$, and $U(C(\xi,\tau^-)$ is the external condition assignment in effect immediately prior to time τ.

Given a trajectory ξ, let $\xi(\tau^-) = (m_{\tau^-}, T_{\tau^-}, clock_{\tau^-})$ indicate the state immediately prior to time τ.

Definition 1 : Given a system $\mathcal{G}$ and external condition assignment law U, a *valid* trajectory ξ satisfies the following for all time τ:

1. T_τ must be both marking enabled and condition enabled under m_{τ^-} and $U(C(\xi,\tau)$, and $clock_{\tau^-}(t) \in [r_{min}(t), r_{max}(t))$.
2. For all $p \in \mathcal{P}_{\mathcal{G}}$,
$$m_\tau(p) = m_{\tau^-}(p) - | \, p^{(t)} \cap T_\tau \, | + | \, {}^{(t)}p \cap T_\tau \, |$$
3. For any transition $t \in \mathcal{T}_{\mathcal{G}}$, if $\{t\}$ is marking enabled and condition enabled under m_τ and u_τ but not under m_{τ^-} and u_{τ^-}, then $clock_\tau(t) = 0$. While t is marking

enabled and condition enabled at time τ, $clock_\tau(t)$ increases with a rate of 1, and $clock_\tau(t) \in [0, r_{max}(t))$. Otherwise, $clock_\tau(t) = -1$.

The set of all valid evolutions for an initial marking m_0 and an external condition assignment policy U is denoted $\Xi(m_0, U)$.

From the above definition, we see that the marking is updated at a time τ by firing a set of transitions T, each of which has been both marking enabled and condition enabled over a period at least $r_{min}(t)$ and less than $r_{max}(t)$. It can thus be considered that upon a transition becoming enabled, a firing delay between $r_{min}(t)$ and $r_{max}(t)$ is chosen, such that t will never be continuously enabled longer than $r_{max}(t)$ without firing (although the transition may become disenabled before its intended firing time, in which case the firing does not happen). The clock for each transition is an indication of how long the transition has been enabled.

3. Layered Control Synthesis

The control problems we consider are specified by sequences, $(C_0, C_1, \ldots C_n) \in (2^{\mathcal{C}})^*$ of condition sets. Given a condition sequence $CSeq = (C_0 C_1 C_2 \ldots C_n)$ and a condition set C, define $CSeq|_C := ((C_0 \cap C)(C_1 \cap C) \ldots (C_n \cap C))$ as the projection of $CSeq$ onto C. Given two condition sequences $CSeq = (C_0 C_1 C_2 \ldots C_n)$ and $CSeq' = (C'_0 C'_1 \ldots C'_{n'})$, we define the order relation $\leq$ such that $CSeq \leq CSeq'$ if there exist integers $0 \leq k_0 \leq k_1 \ldots \leq k_K := n'$ with $k_{-1} := 0$, such that for $0 \leq j \leq n$, $C_j \subseteq C'_i$ for $k_{j-1} \leq i \leq k_j$ when $k_{j-1} = k_j$ and for $k_{j-1} < i \leq k_j$ when $k_{j-1} \neq k_j$.

If $CSeq \leq CSeq'$, then $CSeq'$ has at least as many or more conditions specified in it than $CSeq$. Note that a single condition set C_j in $CSeq$ may correspond to a sequence of condition sets in $CSeq'$, where each C'_i in the sequence includes the conditions in C_j, but may include additional conditions as well. If $CSeq \leq CSeq'$ but $CSeq' \not\leq CSeq$, then $CSeq$ can be thought of as being *less specific* than $CSeq'$. The reverse relation, $\geq$, is defined in a similar manner.

Our control problem can now be stated as follows:

> Given a system $\mathcal{G}$ with initial state $(m_0, \emptyset, clock_0)$ and a goal specification sequence $CSeq$, determine a feedback assignment policy such that for any trajectory $\xi \in \Xi((m_0, \emptyset, clock_0), U)$ there exists a time τ such that $CSeq \leq C(\xi, \tau')$ for all $\tau' \geq \tau$.

Remark: Obvious necessary conditions for the existence of such a policy are that m_0 must satisfy C_0, and for each C_i, $0 \leq i \leq n$, and $c \in C_i$ implies $not(c) \notin C_i$ for each i. The condition sets in the goal sequence may be empty, in which case they are trivially satisfied by any marking and any external condition assignment.

Definition 2 : Let $CSeq = (C_0 C_1 \ldots C_n)$ be a specification. Define control policy U_{CSeq} for a system $\mathcal{G}$ such that for any $\sigma \in (2^{\mathcal{C}})^*$ and any $0 \leq i \leq n$ for which $(C_0 \ldots C_i) \leq \sigma$ and $(C_0 \ldots C_i C_{i+1}) \not\leq \sigma$, then $U_{CSeq}(\sigma) = C_{i+1} - C_{out,\mathcal{G}}$.

Note that it is not specified what the control will be for some σ that is not comparable to a prefix of $CSeq$. Such a control policy in fact does not make sense for most sequences $CSeq$. Such a policy definition is appropriate, however, if $CSeq$ is a *directing sequence.*

Definition 3 : A specification $CSeq$ is a *directing sequence* for a system $\mathcal{G}$ if for all $\xi \in \Xi(\xi(0), U_{CSeq})$, there exists some τ such that for all $\tau' \geq \tau$, $CSeq \leq C(\xi, \tau')$

Lemma 1 : A necessary condition for $CSeq = (C_0 C_1 \ldots C_n)$ to be a directing sequence for $\mathcal{G}$ is that for each i, either $C_{i+1} - C_i \subseteq C_{(out,\mathcal{G})}$ or $(C_{i+1} - C_i) \cap C_{(out,\mathcal{G})} = \emptyset$.

The lemma above follows from the definition of U_{CSeq} by noting that the control law cannot respond to changes in system output before they occur. The policies we consider can thus be characterized as "hold until sense" policies: The controller will set conditions for each i where $(C_{i+1} - C_i) \cap C_{(out,\mathcal{G})} = \emptyset$, until it reaches an i where $(C_{i+1} - C_i) \subseteq C_{(out,\mathcal{G})}$, when it then stops and waits for the system to generate the necessary conditions in $C_{(out,\mathcal{G})}$.

Lemma 2 : Given two condition sequences $CSeq_1$ and $CSeq_2$ such that $CSeq_1$ is a directing sequence for $\mathcal{G}$, if $CSeq_1 \leq CSeq_2$ and $CSeq_2|_{C_{(out,\mathcal{G})}} \leq CSeq_1$, then $CSeq_2$ is also a directing sequence for $\mathcal{G}$.

The above lemma states that $CSeq_2$ will also be a directing sequence if it is at least as specific as $CSeq_1$, but does not impose any further conditions on the output of $\mathcal{G}$.

Our control synthesis approach is to use the above lemma to build up a directing sequence for the system by considering seperately different element nets of the system. The approach is to take a specification $CSeq_0$ for a system $\mathcal{G} = \{G_1 \ldots G_K\}$ and determine a directing sequence $CSeq_1$ for G_1 such that $CSeq_0 \leq CSeq_1$. We then consider G_2 and use $CSeq_1$ as a specification, and derive a directory sequence $CSeq_2$ for G_2. This is repeated until all nets have been considered, at which point, under certain conditions, the resulting $CSeq_K$ is a directing sequence not just for G_K, but for the entire system $\mathcal{G}$. The remainder of this section is devoted to developing the conditions under which this approach is guaranteed to work.

Lemma 3 : Given $CSeq_1$ and state graph net G, there exists a $CSeq_1 \leq CSeq_2$ such that $CSeq_2$ is a directing sequence for G if:

1. $CSeq_1|_{C_{(in,G)}} = \emptyset$;
2. $\exists$ path $\pi = (p_0 t_1 p_1 \ldots t_n p_n)$ in G such that
   (a) $m_0(p_0) = 1$;
   (b) $CSeq_1|_{C_{(out,G)}} \leq (g(p_0)g(p_1)\ldots g(p_n))$;
   (c) for each p_i and all $t \in p_i^{(t)} - t_{i+1}$, $\varphi(t) - \varphi(t_{i+1}) \neq \emptyset$; and
   (d) $\emptyset \neq \varphi(t) \subseteq C_{(in,G)}$ for all $t \in \mathcal{T}_G$

Proof: Item 1 ensures that the $CSeq_1$ doesn't already constrain the inputs of G. Item 2 states that there exists a path in the net such that moving along the path from the initial marking will satisfy $CSeq_1$. Item 2c ensures that an external condition assignment policy will be able to disable transitions that lead away from the path while enabling transitions that lead along the path. Item 2d ensures that each transition can be enabled or disabled exclusively by externally assignable conditions. This is also sufficient for ensuring that the final condition set can be maintained indefinitely. □□□

Let us now demonstrate the construction of a directing sequence for G. Given a sequence $CSeq_1 = (C_1 \ldots C_J)$ for net G such that the conditions in the preceding lemma are satisfied, let $\pi = (p_0 t_1 p_1 \ldots t_n p_n)$ be a path satisfying condition 2b above. There thus exists a strictly increasing sequence of integers $k_0, k_1, k_2, \ldots k_J$ with $k_J := n$, $k_0 := -1$, such that $g(p_i) = C_j$ for $k_{j-1} < i \leq k_j$ for $1 \leq j \leq J$. For each p_i, $i > 0$, define $Precede(p_i) \in \mathcal{C}$ such that $\varphi(t_{i+1}) \subseteq Precede(p_i)$ and for each $t \in p_{i-1}^{(t)} - t_i$ there exists

$c \in Precede(p_i)$ such that $not(c) \in \varphi(t) - \varphi(t_i)$. Thus $Precede(p_i)$ is the set of conditions necessary to enable transition t_i to move a token from p_{i-1} to p_i, while disabling other transitions leading from p_{i-1}. Define $Maintain(p_i)$ such that for all $t \in p_n^{(\iota)}$, there exists some $c \in Maintain(p_i)$ such that $not(c) \in \varphi(t)$. $Maintain(p_i)$ is thus the conditions necessary to keep p_i marked indefinitely (assuming all transitions have nonempty $\varphi(t)$). Note that the sets $Precede(p_i)$ and $Maintain(p_i)$ in general might not be unique. For $i > 0$, we then define:

$$
\begin{aligned}
C_{i,P} &= Precede(p_i) \cup C_{j-1} \quad \text{for } k_{j-1} < i \leq k_j \\
C_{i,S} &= \begin{cases} Precede(p_i) \cup C_{j-1} & \text{for } k_{j-1} < i < k_j \\ Precede(p_i) \cup C_j & \text{for } i = k_j \end{cases} \\
C_{i,M} &= \begin{cases} Precede(p_i) \cup C_{j-1} & \text{for } k_{j-1} < i < k_j \\ Maintain(p_i) \cup C_j & \text{for } i = k_j \end{cases} \\
CSeq_G &= (C_{1,P} C_{1,S} C_{1,M} C_{2,P} C_{2,S} C_{2,M} \ldots C_{n,P} C_{n,S} C_{n,M})
\end{aligned}
$$

We refer to the resulting $CSeq_G$ as a PSM (Precede/Sense/Maintain) condition sequence for G given specification $CSeq$. We see from the above definition that the sequence results in a control that will activate conditions in $Precede(p_i)$ to ensure that the system is enabled to move to a next place p_i but is not enabled to move to some other place. For any p_i with $i \neq k_j$ for any j, then the condition sets $C_{i,P}, C_{i,S}$, and $C_{i,M}$ are equal. However, when $i = k_j$, then $C_{i,S}$ differs from $C_{i,P}$ in conditions which are not externally assigned, i.e. they are sensed instead. A control policy U_{CSeq_G} will wait for this condition set to become true, before implementing the maintaining conditions in $C_{i,M}$.

Lemma 4 : Given a net G and specification $CSeq$ satisfying the conditions of lemma 3, and given the PSM sequence $CSeq_G$ defined from it, then $CSeq \leq CSeq_G$ and $CSeq_G$ is a directing sequence for G.

Proof: From the definition of the sets $C_{i,P}, C_{i,S}, C_{i,M}$, it is clear that $CSeq \leq CSeq_G$. To show that $CSeq_G$ is also a directing sequence, we have to show that any $\xi \in \Xi(\xi(0), U_{CSeq}$ has $CSeq_G \leq C(\xi, \tau)$ for τ chosen large enough. For each p_i, $i > 0$, note that $C_{i,P}$ and $C_{i,S}$ enable only t_i, ensuring that p_i will become marked after p_{i-1}. The condition 2d in lemma 3 ensures that each transition can be enabled or disabled through external condition assignment of the controller. Any ξ thus can only sequence through the path π, finally stopping in place p_n by $Maintain(p_n)$. □□□

Lemma 5 : Given a system $\mathcal{G} = \{G_1, G_2, \ldots G_K\}$ with directing sequence $CSeq_K$, and given a net G_{K+1} satisfying the conditions of lemma 3 for $CSeq_K$, let $CSeq_{K+1}$ be the PSM directing sequence for G_{K+1} given $CSeq_K$. If $C_{(in,G_{K+1})} \cap C_{(out,\mathcal{G})} = \emptyset$, then $CSeq_{K+1}$ is also a directing sequence for $\mathcal{G}$.

Proof: The lemma follows from lemma 2 by noting that the PSM sequence constructed only adds conditions in $C_{(in,G_{K+1})}$, so $CSeq_{K+1}|_{C_{(out,\mathcal{G})}} \leq CSeq_K$. □□□

From the above lemma, $CSeq_{K+1}$ is directing for $\mathcal{G}$ and for G_{K+1}, but will it be directing for $\mathcal{G} \cup \{G_{K+1}\}$? In general, this is not guaranteed. When considering $\mathcal{G}$ alone, $CSeq_{K+1}$ drives the inputs of $\mathcal{G}$ directly and immediately, but when considering $\mathcal{G} \cup \{G_{K+1}\}$, the inputs of $\mathcal{G}$ may be outputs of G_{K+1}, so there can be a delay before an input of $\mathcal{G}$ can be

changed. This delay means some transitions in $\mathcal{G}$ may not be disabled quickly enough to prevent them from firing, thus allowing the system to wander outside the specification.

$CSeq_{K+1}$ can be directing for system $\mathcal{G} \cup \{G_{K+1}\}$ in either of two cases. For the first case, if for any path $(p_0 t_1 p_1)$ in $\mathcal{G}$, if $Precede(p_1)$ and $Maintain(p_1)$ can be chosen such that $c \in Maintain(p_1)$ implies $not(c) \notin Precede(p_1)$, then output transitions to any place p_1 can be disabled before a token enters the place, preventing the problem of disabling the transitions quickly enough after the place has become marked. The PSM sequence definition would need to be modified to enforce $Maintain(p_i)$ before $Precede(p_i)$, and the initial marking would have to be such $Maintain$ conditions are initially true for all initially marked places.

The second case for which $CSeq_{K+1}$ is directing for $\mathcal{G} \cup G_{K+1}$ is if the speed of response of G_{K+1} is significantly faster than the maximum time allowed for enabling and disabling input conditions for $\mathcal{G}$. In other words, external assignments to the inputs of G_{K+1} will alter the truth of conditions in $C_{(out,G_{K+1})} \cap C_{(in,\mathcal{G})}$ rapidly enough to prevent unwanted transition firings. In [6], a method was developed for determining time bounds for how long a condition could be maintained true. These bounds were calculated for the composed system, using worst case situations. If such a bound is determined to be infinity, then the condition can be maintained indefinitely, implying that output transitions for the places can be disabled quickly enough. In such a case, the directing sequence $CSeq_{K+1}$ will be directing for the system $\mathcal{G} \cup \{G_{K+1}\}$. Formalizing the conditions under which this is guaranteed is a topic of current research.

4. Example

We illustrate the method with the following example. Consider the two nets in figure 2. Let the original goal specification be

$$CSeq_0 = (\emptyset\ \{\text{Home}\}\ \emptyset\ \{\text{LBin}\})$$

indicating the requirement that the robot should go to Home and then to LBin. We use a condition label such as Home to indicate the condition (Home, 1) and an overbar such as $\overline{\text{Home}}$ to indicate the condition (Home, 0). The $\emptyset$ conditions in $CSeq_0$ indicate that there may be an initial state before Home is not satisfied, and it is allowable (and in fact necessary) to have a state between where the Home condition is TRUE and the LBin condition is TRUE.

$CSeq_0$ can be used with net G_1 to define a PSM sequence $CSeq_1$ as

$$\begin{aligned} CSeq_1 \;=\; & (\emptyset\ \{\text{MR}, \overline{\text{ML}}\}\ \{\text{MR}, \overline{\text{ML}}, \text{Home}\}\ \{\overline{\text{MR}}, \overline{\text{ML}}, \text{Home}\}\ \{\overline{\text{MR}}, \text{ML}, \} \\ & \{\overline{\text{MR}}, \text{ML}, \text{Lbin}\}\ \{\overline{\text{MR}}, \overline{\text{ML}}, \text{Lbin}\}) \end{aligned}$$

$CSeq_1$ can be used with net G_2 to define a PSM sequence $CSeq_2$:

$$\begin{aligned} CSeq_2 \;=\; & (\emptyset\ \{\text{O2}/1, \overline{\text{O2}/2}\}\{\text{O2}/1, \overline{\text{O2}/2}, \text{MR}, \overline{\text{ML}}\}\ \{\text{O2}/1, \overline{\text{O2}/2}, \text{MR}, \overline{\text{ML}}, \text{Home}\} \\ & \{\overline{\text{O2}/1}, \overline{\text{O2}/2}, \text{MR}, \overline{\text{ML}}, \text{Home}\}\ \{\overline{\text{O2}/1}, \overline{\text{O2}/2}, \overline{\text{MR}}, \overline{\text{ML}}, \text{Home}\} \\ & \{\overline{\text{O2}/1}, \text{O2}/2, \overline{\text{MR}}, \overline{\text{ML}}, \text{Home}\}\{\overline{\text{O2}/1}, \text{O2}/2, \overline{\text{MR}}, \text{ML}, \text{Home}\} \\ & \{\overline{\text{O2}/1}, \text{O2}/2, \overline{\text{MR}}, \text{ML}\}\{\overline{\text{O2}/1}, \text{O2}/2, \overline{\text{MR}}, \text{ML}, \text{Lbin}\} \\ & \{\overline{\text{O2}/1}, \overline{\text{O2}/2}, \overline{\text{MR}}, \text{ML}, \text{Lbin}\}\{\overline{\text{O2}/1}, \overline{\text{O2}/2}, \overline{\text{MR}}, \overline{\text{ML}}, \text{Lbin}\}) \end{aligned}$$

In the above condition sequences, subsequences of identical condition sequences are listed as only one sequence. Note from the definition of the controller policy that sequentially repeated condition sets in a directing condition sequence do not affect the control policy.

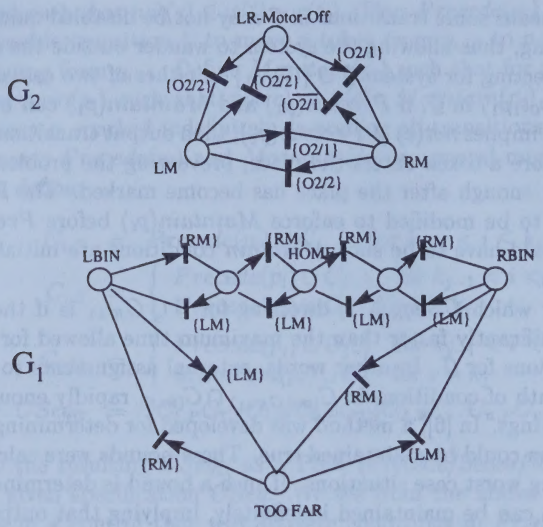


Fig. 2. An example of the robot arm horizontal position and the LR motor state

Filtering out conditions that the controller cannot decide or cannot observe, we get:

$$\begin{aligned} CSeq \;=\; & (\emptyset\ \{\mathrm{O2/1}, \overline{\mathrm{O2/2}}\}\{\mathrm{O2/1}, \overline{\mathrm{O2/2}}, \mathrm{Home}\} \\ & \{\overline{\mathrm{O2/1}}, \overline{\mathrm{O2/2}}, \mathrm{Home}\}\ \{\overline{\mathrm{O2/1}}, \mathrm{O2/2}, \mathrm{Home}\}\ \{\overline{\mathrm{O2/1}}, \mathrm{O2/2}\} \\ & \{\overline{\mathrm{O2/1}}, \mathrm{O2/2}, \mathrm{Lbin}\}\ \{\overline{\mathrm{O2/1}}, \overline{\mathrm{O2/2}}, \mathrm{Lbin}\}) \end{aligned}$$

Note that $CSeq_0 \leq CSeq_1 \leq CSeq_2$. The speed of changing states of the motor is much faster than the speed of the change of states of the robot arm, so the resulting sequence $CSeq_2$ is directing for $\{G_1, G_2\}$.

5. Implementation

The control synthesis method described in this paper has been implemented in Microsoft Visual C++ under Windows 95. The implementation is divided into two parts. The first part is a graphical tool that allows specification of the interconnected element nets and the specification of target states/goal conditions. The relationships between the conditions are represented in a graphical structure called an *s-graph*, with arcs representing the maintain and precede relationships of one system to the next. The tool then outputs an ASCII file that lists the sequence of condition sets grouped, as in the PSM definition, into groups, each with a precede condition set, a sense condition sets, and a maintain condition set. This sequence of condition sets lists only conditions which are observed, either as outputs assigned by the controller or as conditions that are sensed.

The second part of the implementation is a tool to convert the ASCII output file of the control blocks into a program readable by an Allen-Bradley PLC-5 family programmable logic controller. The conversion tool defines segments of ladder logic program for each control block, as well as defining code for initialization of the program and sequencing of the precede/sense/maintain condition triples. Figure 3 illustrates the structure of each

code block. A binary variable, shown as $block_i$, is used to activate a code block. If the variable is TRUE, then the controller has already sequenced through the preceding condition sets (code blocks). A second binary variable, $blockdone_i$, is used to track whether the sensing condition of $C_{S,i}$ has occurred yet. While the sensing is not true, the Precede conditions are latched or unlatched according to $C_{P,i}$. Once the sensing occurs, the Maintain conditions are latched or unlatched according to $C_{M,i}$. The second line of ladder logic is used to reset the binary variables to enable the next control block.

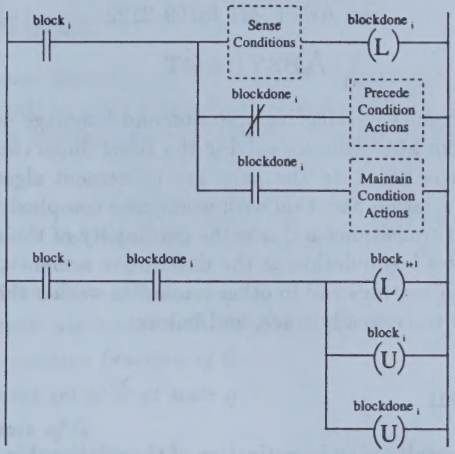


Fig. 3. A PLC code block representing a PSM condition set subsequence.

References

[1] R.S. Sreenivas and B.H. Krogh. On condition/event systems with discrete state realizations. *Discrete Event Dynamic Theory and Applications*, 1(2), September 1991.

[2] Bertil Brandin. P(l)c-based supervisory control of discrete event systems. In *Proceedings 32nd Annual Allerton Conference on Communication, Control, and Computing*, pages 881–889, Monticello, Illinois, 1994.

[3] R.J. Leduc and W.M. Wonham. PLC implementation of a DES supervisor for a manufacturing testbed. In *Proceedings 33rd Annual Allerton Conference on Communication, Control, and Computing*, pages 519–528, Monticello, Illinois, October 1995.

[4] Inger Klein, Peter Jonsson, and Christer Bäckström. Tractable correct planning for an assembly line. In *IFCAI-95 Workshop on Intelligent Manufacturing Systems*, Montreal, 1995.

[5] M. Rausch, A. Lüder, and H-M. Hanisch. Combined synthesis of locking and sequential controllers. In *IEE International Workshop on Discrete Event Systems (WODES96)*, pages 133–138, Edinburgh, August 1996.

[6] L. Holloway. Time measures and state maintainability for a class of composed systems. In *IEE International Workshop on Discrete Event Systems (WODES96)*, pages 24–30, Edinburgh, August 1996.

A Bisimulation Approach to the Supervisory Control of Discrete Event Systems [1]

George Barrett and Stéphane Lafortune
Department of Electrical Engineering and Computer Science
The University of Michigan
1301 Beal Avenue
Ann Arbor, MI 48109–2122

Abstract

We present theoretical results relating bisimulation and language controllability. These results support an efficient algorithm for solving the Basic Supervisory Control Problem (BSCP). The algorithm is based on the partition refinement algorithm of Fernandez, which finds the coarsest relation partition with worst-case complexity $O(m \log(n))$ where m is the total number of transitions and n is the cardinality of the state space. Because the new algorithm utilizes bisimulation as the descriptive semantic, the results may be useful in nondeterministic settings and in other semantics weaker than bisimulation, e.g., ready, trajectory, failure trace, ready trace, and failure.

1 Introduction

This paper presents the results of an investigation of the relationship between supervisory control of DES [10] and bisimulation relations [1, 2]. The approach taken throughout this text is from the point of view of automata as DES models [3]. A basic understanding of DES supervisory control is assumed, for the introductory supervisory material given here is brief. One of the main focal points of this paper is the exploitation of the level of information contained in a finite-state automaton model when solving the supervisory control problem.

Of particular interest is the fact that an automaton model contains not only the language information about a DES but also the branching structure of allowable sequences of events. This level of detail of information allows the supervisory control problem to be generalized to several process-theoretic semantics. A semantics used in computer science which has not previously been applied to the supervisory control problem is the bisimulation semantics. A discussion of bisimulation is given, and results are presented for its application to the supervisory control problem.

This paper presents theoretical results relating bisimulation and controllability which support an efficient algorithm for solving the "Basic Supervisory Control Problem." The algorithm, presented in section 4, is based on a partition refinement algorithm which finds the coarsest relation partition with complexity $O(m \log(n))$ where m is the number of related pairs in the relation and n is the number of elements in the partition. Because the new algorithm utilizes bisimulation as the descriptive semantics, the results can be applied in nondeterministic settings and in weaker semantics, e.g., trajectory model [6], failure, failure trace, ready, ready trace, and trace (language) [2].

The organization of this paper is as follows. Section 2 gives a brief introduction to automata, bisimulation and partition refinement. An efficient algorithm for finding

[1]Research supported in part by NSF grant ECS-9057967. The authors can be reached by email at : grbarret@eecs.umich.edu and stephane@eecs.umich.edu.

bisimulation relations is also discussed. Section 3 discusses the relationship between controllability in the supervisory control problem and bisimulation. Section 4 discusses using bisimulation to solve the Basic Supervisory Control Problem with both off-line and on-line strategies. The paper concludes with section 5 where the basic ideas are summarized.

2 Preliminaries: Automata, Bisimulation Relations and Partitions

For the investigation and discussion of discrete event systems at a logical level of abstraction, the automaton will be used as the primary descriptive model.

Definition 2.1 [3] *A* **deterministic finite-state automaton (DFSA)**, *denoted G, is a six-tuple*

$$G = (X_G, \Sigma, \delta_G, \Sigma_G, x_{G0}, X_{Gm})$$

that generates $L = \mathcal{L}(G)$ and marks $L_m = \mathcal{L}_m(G)$ where

$$\begin{aligned} X_G &= \textit{finite set of states of } G \\ \Sigma &= \textit{set of events associated with the transitions in } G \\ \delta_G &= \textit{partial transition function of } G \\ \Sigma_G(q) &= \textit{active event set of } G \textit{ at state } q \\ x_{G0} &= \textit{initial state of } G \\ X_{Gm} &= \textit{a subset of } X_G \textit{ which represent marked states. If } L = L_m, \textit{ then } X_{Gm} = X_G. \end{aligned}$$

The transition function δ_G is not necessarily total as in the "standard" definition of automaton, and all automata in this paper are assumed to be accessible, i.e., all unreachable states have been deleted from the model. If X_{Gm} is omitted, then it understood that $X_{Gm} = X_G$. Furthermore, it is common to define the empty trace ϵ to be the trace containing no events. The transition function, δ_G, is often extended from events to traces recursively:

$$\delta_G(x_G, \epsilon) = x_G,$$

and for $t \in \Sigma^*$, $\sigma \in \Sigma$:

$$\delta_G(x_G, t\sigma) = \delta_G(\delta_G(x_G, t), \sigma).$$

If G is a nondeterministic finite-state automaton, then $\delta_G(x_G, t\sigma)$ is a set.

In what follows, $x \xrightarrow{\sigma} x'$ denotes that there exists a transition with the label σ from state x to state x'.

Definition 2.2 *Let G be as above and let $H = (X_H, \Sigma, \delta_H, \Sigma_H, x_{H0}, X_{Hm})$.* ***A* bisimulation relation** *is a binary relation, Φ, between subsets of states $S_H \subseteq X_H$ and $S_G \subseteq X_G$ with respect to a set of events Σ_R satisfying [1, 2]:*

1. *For each $x_H \in S_H$ there exists $x_G \in S_G$ such that $\Phi(x_H, x_G)$.*
2. *For each $x_G \in S_G$ there exists $x_H \in S_H$ such that $\Phi(x_H, x_G)$.*
3. *If $\Phi(x_H, x_G)$, $\sigma \in \Sigma_R$ and $x_H \xrightarrow{\sigma} x'_H$, then there is a x'_G such that $x_G \xrightarrow{\sigma} x'_G$ and $\Phi(x'_H, x'_G)$.*
4. *If $\Phi(x_H, x_G)$, $\sigma \in \Sigma_R$ and $x_G \xrightarrow{\sigma} x'_G$, then there is a x'_H such that $x_H \xrightarrow{\sigma} x'_H$ and $\Phi(x'_H, x'_G)$.*
5. *(Bisimulation with marking) $\Phi(x_H, x_G)$ implies $x_H \in X_{Hm}$ iff $x_G \in X_{Gm}$.*

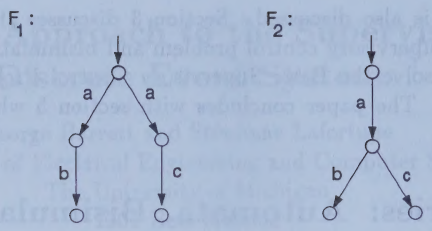


Figure 1: Example of simulation relation

Bisimulation relations are closed under arbitrary unions [1, 2], so there exists a *maximal* bisimulation relation. Automata H and G are *bisimilar*, denoted $H \leftrightarrows G$, if there exists a bisimulation relation over all of X_H and X_G, and (x_{H0}, x_{G0}) is in the relation. If ψ is a relation or set of state pairs of X_H and X_G, denote by $H \leftrightarrows_\psi G$ if ψ is a bisimulation relation; for example, let Φ be the bisimulation relation with respect to "a" between F_1 and F_2 of Figure 1, then $H \leftrightarrows_\Phi G$, but $H \not\leftrightarrows G$. Bisimulation is an equivalence relation between sets of states for a given set of transitions; that is, two bisimilar states have equivalent future transitions. An equivalence relation over a set is a partition of that set, hence a bisimulation relation between H and G can be represented by a partitioning of the set of states of $X_H \cup X_G$. In the sequel, denote by Φ_Π the *maximal* bisimulation relation induced by the partition Π where the set of states and events are understood by the context.

Definition 2.3 *Let H and G be as above. A* **simulation relation** *is a binary relation, R_S, between a subset, S_H, of the states of H and a subset, S_G, of the states of G and with respect to a set of events Σ_R, satisfying [1]:*

1. *For each $x_H \in S_H$ there exists $x_G \in S_G$ such that $R_S(x_H, x_G)$.*
2. *If $R_S(x_H, x_G)$, $\sigma \in \Sigma_R$ and $x_H \xrightarrow{\sigma} x'_H$, then there is a x'_G such that $x_G \xrightarrow{\sigma} x'_G$ and $R_S(x'_H, x'_G)$.*
5. *(Simulation with marking) $R_S(x_H, x_G)$ implies $x_H \in X_{Hm}$ iff $x_G \in X_{Gm}$.*

If $R_S(x_H, x_G)$ then x_G *simulates* x_H. Automaton H is *simulated* by automaton G, denoted $H \sqsubseteq_{SR} G$, if there exists a simulation relation R_S such that every state in X_H is simulated by a state in X_G, and $R_S(x_{H0}, x_{G0})$. Clearly, if H and G are deterministic, then H is simulated by G iff $\mathcal{L}(H) \subseteq \mathcal{L}(G)$ and $\mathcal{L}_m(H) \subseteq \mathcal{L}_m(G)$.

As an example of simulation relation, consider Figure 1. Automaton F_1 is simulated by automaton F_2; however, F_2 is not simulated by F_1. Notice that $\mathcal{L}(F_1) = \mathcal{L}(F_2)$. This example also shows the inability of language to capture nondeterministic behavior described by the branching structure of finite-state automata. For example, after "a" occurs F_1 can be at a state where "c" can never occur, but "c" can always occur following "a" in F_2.

Definition 2.4 *Let G be a finite state automaton (not necessarily deterministic) whose maximal bisimulation relation with itself is Φ. Denote by C the set of equivalence classes induced by Φ. The* **normal form** *of G, denoted by $N(G)$, is the automaton generated by using the equivalence classes of Φ as the states of $N(G)$ [2]. The states of $N(G)$ inherit marking information and all incoming and outgoing event transitions from the previous states of G. $N(G)$ can be called the minimum state realization of G.*

In the trace or language semantics, two discrete event systems modeled by DFSA A_1 and A_2, are considered equivalent if they have equivalent trace sets, i.e., $\mathcal{L}(A_1) = \mathcal{L}(A_2)$ and $\mathcal{L}_m(A_1) = \mathcal{L}_m(A_2)$. The bisimulation semantics, however, has a more detailed requirement for DES equivalence. Within the bisimulation semantics two DES modeled by automata A_1 and A_2, are equivalent if their *normal forms* are equal (isomorphic), i.e., $N(A_1) = N(A_2)$ [2]. Figure 1 shows a case where two automata may be equivalent in the trace semantics and not equivalent in the bisimulation semantics. If A_1 and A_2 are both deterministic, then $N(A_1) = N(A_2)$ iff $\mathcal{L}(A_1) = \mathcal{L}(A_2)$ and $\mathcal{L}_m(A_1) = \mathcal{L}_m(A_2)$.

Despite bisimulation being finer than trace equivalence, algorithms exist for bisimulation equivalence that are more efficient than analogous algorithms for regular languages. If the automata being processed by these algorithms are deterministic, then the bisimulation results are equivalent to trace results. Furthermore, if the automata being processed are nondeterministic, then the bisimulation results are only sufficient for trace results.

In [9], Paige and Tarjan present an efficient algorithm which solves the *relational coarsest partition problem*: given a partition Π_1 of a set U and a binary relation E on U, find the coarsest refinement Π_2 of Π_1 such that for each pair of blocks B_1, B_2 of Π_2, either $B_1 \subseteq E^{-1}(B_2)$ or $B_1 \cap E^{-1}(B_2) = \emptyset$. $E^{-1}(B_2)$ is the *preimage* set: $E^{-1}(B_2) = \{x | \exists y \in B_2$ such that $E(x, y)\}$.

The algorithm of Paige and Tarjan considers the case of a single relation, i.e., $|\Sigma| = 1$, thus limiting its usefulness on systems with multiple events in their alphabet. Fernandez [4] generalizes the algorithm of Paige and Tarjan to handle the case when $|\Sigma| \geq 1$. When applied to two automata, A_1 and A_2, the algorithm of Fernandez runs in time bounded by[2] $(m_1 + m_2)\log(n_1 + n_2)$ and space bounded by $(m_1 + m_2 + n_1 + n_2)$ where m_1 and m_2 are the number of transitions in A_1 and A_2, and n_1 and n_2 are the number of states in A_1 and A_2, respectively.

As stated earlier, a bisimulation relation is an equivalence relation making it representable as a partition. It follows then that the problem of finding the coarsest partition that is *stable* with respect to a relation E is the same problem as finding the maximal bisimulation relation with respect to E [1, 4].

3 Controllability and Bisimulation

This section presents the main theoretical results of this paper that relate controllability with bisimulation. These results support efficient algorithms that will be discussed later for solving supervisory control and related problems.

Theorem 3.1 *Given two deterministic finite state automata* $A_1 = (Q_1, \Sigma, \delta_1, \Sigma_1, q_{0,1})$ *and* $A_2 = (Q_2, \Sigma, \delta_2, \Sigma_2, q_{0,2})$ *where* $\mathcal{L}(A_1) \subseteq \mathcal{L}(A_2)$, *let* $\psi \subseteq Q_1 \times Q_2$ *be the set of state pairs that are reachable by traces in* $\mathcal{L}(A_1)$. *Further, let* $\Phi_{uc} \subseteq Q_1 \times Q_2$ *be the maximal bisimulation relation between* A_1 *and* A_2 *with respect to* Σ_{uc}. $\mathcal{L}(A_1)$ *is controllable with respect to* $\mathcal{L}(A_2)$ *and a set,* $\Sigma_{uc} \subseteq \Sigma$, *of uncontrollable events if and only if* $\psi \subseteq \Phi_{uc}$, *i.e.,* ψ *is a bisimulation relation of* A_1 *and* A_2 *with respect to* Σ_{uc}.

Proof 3.1 *Since* Φ_{uc} *is the maximal bisimulation relation with respect to* Σ_{uc}, ψ *is a subset of* Φ_{uc} *iff* ψ *is a bisimulation relation with respect to* Σ_{uc}. *By the language inclusion*

[2]More specifically, the worst-case running time of the Fernandez algorithm is $O(c\, m \log(n))$ where c is a homogeneous upper bound on the number of times any label appears at any state, i.e., c is the largest image set size due to any event [4]. Here, the algorithm is only used on *deterministic* automata so $c = 1$. The text [1] incorrectly states the complexity as $O(mn)$.

assumption and the discussion of Section 2, $A_1 \sqsubseteq_{SR} A_2$ which also implies $A_1 \sqsubseteq_{SR|_{\Sigma_{uc}}} A_2$ (A_1 is simulated by A_2 with respect to Σ_{uc}); hence, ψ is a simulation relation with respect to Σ_{uc}. Using contraposition for simulation in the other direction yields:
if $\mathcal{L}(A_1) \subseteq \mathcal{L}(A_2)$, *then* $\mathcal{L}(A_1)\Sigma_{uc} \cap \mathcal{L}(A_2) \not\subseteq \mathcal{L}(A_1)$
$\Leftrightarrow \exists t \in \mathcal{L}(A_1), \exists \sigma_{uc} \in \Sigma_{uc}$ *such that* $t\sigma_{uc} \notin \mathcal{L}(A_1) \wedge t\sigma_{uc} \in \mathcal{L}(A_2)$.
$\Leftrightarrow \exists s_1 = \delta_1(q_{0,1}, t), \exists s_2 = \delta_2(q_{0,2}, t)$ *such that* $\sigma_{uc} \notin \Sigma_{uc}(s_1) \wedge \sigma_{uc} \in \Sigma_{uc}(s_2)$
$\Leftrightarrow \exists s_1 = \delta_1(q_{0,1}, t), \exists s_2 = \delta_2(q_{0,2}, t)$ *such that* $\Sigma_{uc}(s_2) \not\subseteq \Sigma_{uc}(s_1)$
$\Leftrightarrow \exists s_1 = \delta_1(q_{0,1}, t), \exists s_2 = \delta_2(q_{0,2}, t)$ *such that* $s_2 \not\sqsubseteq_{SR|_{\Sigma_{uc}}} s_1$
$\Leftrightarrow \psi \not\subseteq \Phi_{uc}$ ↓

The importance of ψ is derived from the need to be able to determine if the bisimulation is violated by reachable state pairs or if the bisimulation is violated by unreachable state pairs. In the latter case, the violation is not of consequence.

Definition 3.1 *Let a DES be modeled by the DFSA $G = (X_G, \Sigma, \delta_G, \Sigma_G, x_{G0})$, and let a desired behavior be modeled by the DFSA $H = (X_H, \Sigma, \delta_H, \Sigma_H, x_{H0})$. Denote by $[H \times G]^\uparrow$ the automaton formed using the standard algorithm [12] for computing the supremal controllable sublanguage of $\mathcal{L}(H)$ with respect to $\mathcal{L}(G)$ and Σ_{uc}, i.e., forming the product $H \times G$ and iteratively pruning states which 1) violate the controllability condition or 2) are unreachable; hence, $\mathcal{L}([H \times G]^\uparrow) = \mathcal{L}(H \times G)^\uparrow$.*

Corollary 3.1 *Let H and G be as in Definition 3.1 such that $\mathcal{L}(H) \subseteq \mathcal{L}(G)$. Let ψ be the reachable subspace of $X_H \times X_G$. Then $\mathcal{L}(H)$ is controllable with respect to $\mathcal{L}(G)$ and Σ_{uc} if and only if ψ is a bisimulation relation of H and G with respect to Σ_{uc}, i.e., $H \leftrightarrows_{\psi|_{\Sigma_{uc}}} G$.*

Theorem 3.2 *Let G and H be as in Definition 3.1 such that $\mathcal{L}(H) \subseteq \mathcal{L}(G)$ and $\mathcal{L}(H)^\uparrow = \mathcal{L}([H \times G]^\uparrow)$. Denote the (accessible) state space of $[H \times G]^\uparrow$ by $S^\uparrow$. Let the set of uncontrollable events be Σ_{uc}, and let Φ_{uc} be the maximal bisimulation relation between H and G with respect to Σ_{uc}. Then $(x_H, x_G) \in S^\uparrow$ iff $\Phi_{uc}(x_H, x_G)$ and (x_H, x_G) is reachable from (x_{H0}, x_{G0}) by a sequence of state transitions that never leave Φ_{uc}.*

Proof 3.2 *Let $R = (X_R, \Sigma, \delta_R, \Sigma_R, x_{R0})$ where*

$$
\begin{aligned}
X_R &= \{x_R \in X_H \times X_G | x_H \in X_H, x_G \in X_G, x_R = (x_H, x_G) \in \Phi_{uc}\} \\
x_{R0} &= (x_{H0}, x_{G0}) \\
\forall \sigma \in \Sigma &: \\
\delta_R((x_H, x_G), \sigma) &= \begin{cases} (\delta_H(x_H, \sigma), \delta_G(x_G, \sigma)) & \text{if } (\delta_H(x_H, \sigma), \delta_G(x_G, \sigma)) \in X_R \\ \text{undefined} & \text{otherwise} \end{cases} \\
\Sigma_R((x_H, x_G)) &= \{\sigma \in \Sigma | \delta_R((x_H, x_G), \sigma) \text{ is defined}\}.
\end{aligned}
$$

Define $X_R^\uparrow$ to be the reachable state space of R; hence, $X_R^\uparrow$ is the maximal bisimulation relation between H and G with respect to Σ_{uc} that is reachable from (x_{H0}, x_{G0}). For $(x_H, x_G) \in X_R^\uparrow$, Definition 2.2 implies:

1. $\forall \sigma_{uc} \in \Sigma_{uc}$ *if* $x_H \xrightarrow{\sigma_{uc}} x'_H$, *then* $\exists x'_G$ *such that* $x_G \xrightarrow{\sigma_{uc}} x'_G$ *and* $(x'_H, x'_G) \in X_R^\uparrow$
2. $\forall \sigma_{uc} \in \Sigma_{uc}$ *if* $x_G \xrightarrow{\sigma_{uc}} x'_G$, *then* $\exists x'_H$ *such that* $x_H \xrightarrow{\sigma_{uc}} x'_H$ *and* $(x'_H, x'_G) \in X_R^\uparrow$.

Now, let $(\tilde{x}_H, \tilde{x}_G) \in S^\uparrow$. By language inclusion:

$$\forall \sigma_{uc} \in \Sigma_{uc} \text{ if } \tilde{x}_H \xrightarrow{\sigma_{uc}} \tilde{x}'_H, \text{ then } \exists \tilde{x}'_G \text{ such that } \tilde{x}_G \xrightarrow{\sigma_{uc}} \tilde{x}'_G \text{ and } (\tilde{x}'_H, \tilde{x}'_G) \in S^{\uparrow}.$$

By controllability:

$$\forall \sigma_{uc} \in \Sigma_{uc} \text{ if } \tilde{x}_G \xrightarrow{\sigma_{uc}} \tilde{x}'_G, \text{ then } \exists \tilde{x}'_H \text{ such that } \tilde{x}_H \xrightarrow{\sigma_{uc}} \tilde{x}'_H \text{ and } (\tilde{x}'_H, \tilde{x}'_G) \in S^{\uparrow}.$$

Hence, $S^{\uparrow} = X_R^{\uparrow}$, and by the construction of $X_R^{\uparrow}$, (x_H, x_G) state pairs can only be members of $S^{\uparrow}$ if they are reachable from (x_{H0}, x_{G0}) by sequences of state transitions that remain in Φ_{uc}. ⊣

Corollary 3.2 *Let H and G be as in Theorem 3.2, and let R be as in Proof 3.2, then $\mathcal{L}(H)^{\uparrow} = \mathcal{L}(R)$.*

The maximal bisimulation relation between H and G with respect to Σ_{uc}, Φ_{uc}, can be found by partition refinement in time bounded by $O(|\Sigma_{uc}|(n_H + n_G)\log(n_H + n_G))$ [4], where n_H and n_G are the number of states in H and G respectively. Perhaps limiting the importance of this result is that a reachability test is still required to eliminate extraneous unreachable states, i.e., forming the maximal bisimulation relation in a running time of $O(|\Sigma_{uc}|(n_H + n_G)\log(n_H + n_G))$ does not allow controllability to be determined in '$m\log(n)$', too. This issue is currently being investigated; however, for many applications (such as constructing on-line supervisors) it is often the case that reachability is not of concern.

4 Solution to the Supervisory Control Problem using Bisimulation

4.1 Off-Line Solution to the BSCP

Let a DES and desired behavior be modeled as in Definition 3.1 by G and H where $\mathcal{L}(H) \subseteq \mathcal{L}(G)$ and $\mathcal{L}(H)^{\uparrow} = \mathcal{L}([H \times G]^{\uparrow})$. Let every state be marked in G and H, and let the set of uncontrollable events be Σ_{uc}. The problem of calculating $\mathcal{L}(H)^{\uparrow}$ in this case is often called (see, e. g. [3]) the Basic Supervisory Control Problem (or BSCP) of supervisory control theory.

Presented here is an algorithm, BISIM-BSCP, that uses bisimulation to find a DFSA that generates $\mathcal{L}(H)^{\uparrow}$.

Off-Line Algorithm: BISIM-BSCP

Step 1. Find the coarsest stable relational partition, Π, between G and H with respect to the set of uncontrollable events, Σ_{uc}.

Step 2. If (x_{H0}, x_{G0}) is not in the equivalence relation represented by Π, then let R be the empty automaton. Otherwise, let $R = (X_R, \Sigma, \delta_R, \Sigma_R, x_{R0})$ where

$$\begin{aligned}
X_R &= \{x_R \in X_H \times X_G | x_H \in X_H, x_G \in X_G, x_R = (x_H, x_G) \in \Phi_\Pi\} \\
x_{R0} &= (x_{H0}, x_{G0}) \\
\forall \sigma \in \Sigma &: \\
\delta_R((x_H, x_G), \sigma) &= \begin{cases} (\delta_H(x_H, \sigma), \delta_G(x_G, \sigma)) & \text{if } (\delta_H(x_H, \sigma), \delta_G(x_G, \sigma)) \in X_R \\ \text{undefined} & \text{otherwise} \end{cases} \\
\Sigma_R((x_H, x_G)) &= \{\sigma \in \Sigma | \delta_R((x_H, x_G), \sigma) \text{ is defined}\}.
\end{aligned}$$

Step 3. Let $R^{\uparrow} = \text{trim}(R)$.

Theorem 4.1 *The automaton $R^{\uparrow}$ generated by following the algorithm* BISIM-BSCP *generates the supremal controllable sublanguage of $\mathcal{L}(H)$ with respect to $\mathcal{L}(G)$ and Σ_{uc}. If (x_{H0}, x_{G0}) is not in the equivalence relation represented by the partition Π, then R is the empty automaton and $\mathcal{L}(H)^{\uparrow} = \emptyset$. Otherwise, the partition Π represents the maximal bisimulation relation between G and H with respect to Σ_{uc} and $\mathcal{L}(R^{\uparrow}) = \mathcal{L}(H)^{\uparrow}$.*

Proof 4.1 *Follows from the results of Section 3.* ⊣

4.2 Solving the Nonblocking Version of BSCP

The problem of calculating $\mathcal{L}_m(H)^{\uparrow}$ in this case is often called (see, e. g. [3]) the Nonblocking version of the Basic Supervisory Control Problem (or BSCP-NB). This problem can also be solved using bisimulation relations by iterating on steps 1-3 of the above algorithm. Let a DES be modeled by the DFSA $G = (X_G, \Sigma, \delta_G, \Sigma_G, x_{G0}, X_{Gm})$, and let a desired behavior be modeled by the DFSA $H = (X_H, \Sigma, \delta_H, \Sigma_H, x_{H0}, X_{Hm})$. Considered here is the case when $X_{Gm} \neq X_G$ and $X_{Hm} \neq X_H$. Assume that $\mathcal{L}_m(H) \subseteq \mathcal{L}_m(G)$ and $\mathcal{L}(H) \subseteq \mathcal{L}(G)$. Let $\mathcal{L}_m(H)^{\uparrow} = \mathcal{L}_m([H \times G]^{\uparrow})$, and let the set of uncontrollable events be Σ_{uc}. The problem of calculating $\mathcal{L}_m(H)^{\uparrow}$ in this case is often called (see, e. g. [3]) the Nonblocking version of the Basic Supervisory Control Problem (or BSCP-NB). This problem can also be solved using bisimulation relations by iterating on steps 1-3 of the above algorithm.

Off-Line Non-Blocking Algorithm: BISIM-BSCP-NB

Step 0. Initialize such that

$$R^{0\uparrow} = \text{the empty automaton}, G^0 = G, H^0 = H, \Pi^0 = \{X_H \cup X_G\}, i = 1.$$

Step 1. Let $G^i = (X_G^i, \Sigma, \delta_G^i, \Sigma_G, x_{G0}, X_{Gm}^i)$, where

$$\begin{aligned} X_G^i &= \{x_G \in X_G | \exists x_H \in X_H \text{ such that } (x_H, x_G) \in \Phi_{\Pi^{i-1}}\} \\ \delta_G^i(x_G, \sigma) &= \begin{cases} \delta_G(x_G, \sigma) & \text{if } \delta_G(x_G, \sigma) \in X_G^i \\ \text{undefined} & \text{otherwise} \end{cases} \\ X_{Gm}^i &= X_{Gm} \cap X_G^i. \end{aligned}$$

Let $H^i = (X_H^i, \Sigma, \delta_H^i, \Sigma_H, x_{H0}^i, X_{Hm}^i)$, where

$$\begin{aligned} X_H^i &= \{x_H \in X_H | \exists x_G \in X_G \text{ such that } (x_H, x_G) \in \Phi_{\Pi^{i-1}}\} \\ \delta_H^i(x_H, \sigma) &= \begin{cases} \delta_H(x_H, \sigma) & \text{if } \delta_H(x_H, \sigma) \in X_H^i \\ \text{undefined} & \text{otherwise} \end{cases} \\ X_{Hm}^i &= X_{Hm} \cap X_H^i. \end{aligned}$$

Find the coarsest stable relational partition, Π^i, between G^i and H^i with respect to the set of uncontrollable events, Σ_{uc}.

Step 2. If (x_{H0}, x_{G0}) is not in the equivalence relation represented by Π^i, then let R^i be the empty automaton. Otherwise, let $R^i = (X_R^i, \Sigma, \delta_R^i, \Sigma_R^i, x_{R0}, X_{Rm}^i)$ where

$$\begin{aligned} X_R^i &= \{x_R \in X_H^i \times X_G^i | x_H \in X_H^i, x_G \in X_G^i, x_R = (x_H, x_G) \in \Phi_{\Pi^i}\} \\ x_{R0} &= (x_{H0}, x_{G0}) \\ \forall \sigma \in \Sigma &: \\ \delta_R^i((x_H, x_G), \sigma) &= \begin{cases} (\delta_H^i(x_H, \sigma), \delta_G^i(x_G, \sigma)) & \text{if } (\delta_H^i(x_H, \sigma), \delta_G^i(x_G, \sigma)) \in X_R^i \\ \text{undefined} & \text{otherwise} \end{cases} \\ \Sigma_R^i((x_H, x_G)) &= \{\sigma \in \Sigma | \delta_R^i((x_H, x_G), \sigma) \text{ is defined}\}. \end{aligned}$$

Step 3. Let $R^{i\uparrow} = \text{trim}(R^i)$. If $R^{i\uparrow} = R^{i-1\uparrow}$, then stop; otherwise go to **Step 1.**

Theorem 4.2 *The automaton $R^{i\uparrow}$ generated at convergence following the algorithm* BISIM-BSCP-NB *generates the supremal controllable sublanguage of $\mathcal{L}_m(H)$ with respect to $\mathcal{L}(G)$ and Σ_{uc}. If $R^{i\uparrow}$ is the empty automaton, then $\mathcal{L}_m(H)^\uparrow = \emptyset$; otherwise $\mathcal{L}_m(R^{i\uparrow}) = \mathcal{L}_m(H)^\uparrow$.*

Proof 4.2 *Follows from the results of Section 3 and the standard algorithm [12] for solving the BSCP-NB.* ⊣

4.3 BSCP Solution Example

This subsection demonstrates the off-line solution of the Basic Supervisory Control Problem using the maximal bisimulation with respect to the set of uncontrollable events. An automaton that represents a desired language is shown in Figure 2. The plant automaton is shown in Figure 3. Assume $\Sigma_{uc} = \{\sigma_4\}$.

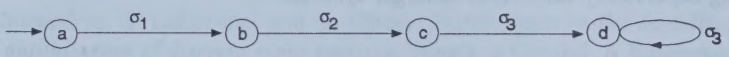


Figure 2: H, Automaton representing desired language

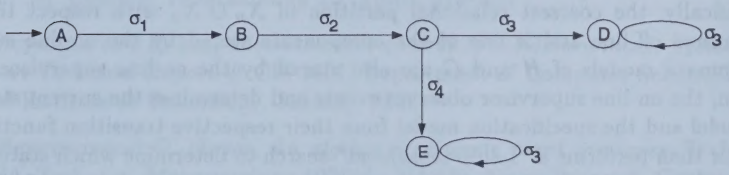


Figure 3: G, Automaton representing plant

To find the coarsest stable relational partition with respect to Σ_{uc}, we begin by forming a partition, Π_0, of the union of all the states in H and G:

$$\Pi_0 = \{I\} = \{\{A, B, C, D, E, a, b, c, d\}\}$$

and determine the preimage set of block I due to the set of uncontrollable events to be:

$$E^{-1}(\{A, B, C, D, E, a, b, c, d\}) = \{C\}.$$

Refining the initial partition with respect to $E^{-1}(I)$ yields the partition:

$$\Pi_1 = \{II, III\} = \{\{A, B, D, E, a, b, c, d\}, \{C\}\}$$

which is stable with respect to both $E^{-1}(II)$ and $E^{-1}(III)$ causing the partition refinement algorithm to terminate. $R^\uparrow$ is the reachable part of the product machine over possible state pairs in Π_1 (cf. step 3 of BISIM-BSCP); it is shown in Figure 4.

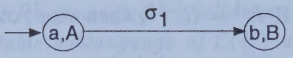


Figure 4: Solution to BSCP

4.4 Discussion of Computational Complexity

Off-Line

Inspection of the off-line solution to the Basic Supervisory Control Problem reveals that the worst-case complexity, $O(|\Sigma|n_H n_G)$, is the same as previous methods for computing the supremal controllable sublanguage when the marked language is prefix closed. However, when the marked language is not prefix closed, using the Fernandez' generalization [4] of the efficient algorithm of Paige and Tarjan [9] to find the maximal bisimulation in $O(|\Sigma_{uc}|(n_H + n_G)\log(n_H + n_G))$ allows step 1 of each iteration of BISIM-BSCP-NB to be lower in complexity than the equivalent step of the standard algorithm which has worst-case complexity of $O(|\Sigma|n_H n_G)$. The improved algorithm may be more practical for finding supervisory controllers of larger systems.

On-Line

If one has a small amount of computing power available for on-line computations, then the reachability test of step 3 of BISIM-BSCP can be avoided if the supervisor is generated on-line. This results in off-line computations of much lower complexity.

Specifically, the coarsest relational partition of $X_H \cup X_G$ with respect to Σ_{uc}, Π, is determined off-line and is one of the components stored by the on-line supervisor. The automata models of H and G are also stored by the on-line supervisor. During operation, the on-line supervisor observes events and determines the current state in the plant model and the specification model from their respective transition functions. The supervisor then performs a "1-step-lookahead" search to determine which state pairs are immediately reachable by controllable events; this operation is done in $O(|\Sigma_c|)$ time. For each pair of possible next states (x'_H, x'_G), if x'_H and x'_G are not in the same block of Π, then the event leading to (x'_H, x'_G) is disabled. Checking a state pair against Π requires $O(1)$ time. In summary, off-line calculations for the on-line supervisor require $O(|\Sigma_{uc}|(n_H+n_G)\log(n_H+n_G))$ time, and the on-line calculations are of complexity $O(|\Sigma_c|)$ at each state. On-line storage is $O(|\Sigma|(n_H + n_G))$.

5 Conclusions

In this paper a different semantics, bisimulation, was used to solve the Basic Supervisory Control Problem in deterministic settings. It was shown that, given language inclusion, controllability of languages is the same as bisimulation with respect to uncontrollable events.

Because bisimulation is an equivalence relation, bisimulation relations can be found by partition refinement for which there exist very efficient algorithms. One such algorithm by Fernandez [4], based on Paige and Tarjan's famous algorithm [9], finds the coarsest partition of a family of binary relations on a set, S, with complexity $O(c\, m \log(n))$ where m is the size of the relation, n is the size of S, and c is a bound on the largest image set of any element of the relation. This algorithm is used in a very efficient on-line solution of the BSCP. Given two automata H and G, the "core" bisimulation relation, Φ, with

respect to Σ_{uc} is found in $O(|\Sigma_{uc}|(n_H + n_G)\log(n_H + n_G))$ time and $O(n_H + n_G)$ space. The on-line supervisor maintains only a copy of G, H, and Φ and performs $O(|\Sigma_c|)$ "1-step-lookahead" operations at each state. Thus, the size of the on-line supervisor is $O(|\Sigma|(n_H + n_G))$.

Supervisory Control Theory bases equivalence on languages, and therefore, must be altered to handle nondeterministic settings; recent work in that regard can be found in [5, 7, 8, 11]. Solving supervisory control problems in the bisimulation semantics provides only sufficient conditions for solutions in weaker semantics; however, this approach may be more practical for larger systems.

References

[1] André Arnold, *Finite transition systems*, Prentice Hall, NJ, 1994.

[2] J. C. M. Baeten and W. P. Weijland, *Process algebra*, Cambridge Tracts in Theoretical Computer Science, vol. 18, Cambridge University Press, Great Britain, 1990.

[3] C. Cassandras, S. Lafortune, and G. Olsder, *Introduction to the modelling, control and optimization of discrete event systems*, Trends in Control. A European Perspective (A. Isidori, ed.), Springer–Verlag, September 1995, pp. 217–291.

[4] J. Fernandez, *An implementation of an efficient algorithm for bisimulation equivalence*, Sci. Comput. Programming **13** (1990), 219–236.

[5] M. Heymann and F. Lin, *Discrete event control of nondeterministic systems*, Tech. Report Technical Report # CIS 9601, Department of Computer Science Technion, Israel Institute of Technology, January 1996.

[6] M. Heymann and G. Meyer, *An algebra of discrete event processes*, Tech. Report NASA Technical Memorandum 102848, NASA, Ames Research Center, Moffett Field, CA, USA, June 1991.

[7] K. Inan, *Nondeterministic supervision under partial observation*, 11th International Conference on Analysis and Optimization of Systems: Discrete Event Systems (G. Cohen and J. Quadrat, eds.), Springer–Verlag, June 1994, pp. 39–48.

[8] A. Overkamp, *Control of nondeterministic discrete event systems using failure semantics*, Proc. of 3rd European Control Conference (New Orleans, LA, USA), December 1995, pp. 422–426.

[9] R. Paige and R. Tarjan, *Three partition refinement algorithms*, SIAM J. Control Optim. **16** (1987), no. 6, 973–989.

[10] P.J. Ramadge and W.M. Wonham, *The control of discrete event systems*, Proc. of the IEEE **77** (1989), no. 1, 81–98.

[11] M. Shayman and R. Kumar, *Supervisory control of nondeterministic systems with driven events via prioritized synchronization and trajectory models*, SIAM J. Control Optim. **33** (1995), no. 2, 469–497.

[12] W.M. Wonham and P.J. Ramadge, *On the supremal controllable sublanguage of a given language*, SIAM J. Control Optim. **25** (1987), no. 3, 637–659.

Control of Timed Discrete-Event Systems — A Framework

Tan-Jan Ho

Systems Control Group
Department of Electrical and Computer Engineering
University of Toronto
Toronto, CANADA

e-mail: tjho@odin.control.utoronto.ca

Abstract

In this paper, a framework for the control of timed discrete-event systems is developed. First of all, an untimed modelling formalism is proposed. It incorporates data variables and is an extension of an automata model. Second, a timed model is established based on the untimed model . The newly-proposed timed model is a simplified version of the framework proposed by Ho and Wonham [2]. It preserves the modelling merit of TTM's [4]. For control of timed discrete-event models, timed state-feedback control mechanisms will be introduced .

Examples are provided to illustrate how to use the proposed timed model as the formal structure to describe the constituent subsystems and the overall system. We also offer an example to demonstrate the advantage over TTM's and timed automata in [1] by employing the setup in timed systems design.

1 Introduction

The operation of a discrete-event system (DES) is modeled as an automata model in a series of papers [5], [6] and [7] initiated by Ramadge and Wonham. In this paper, we introduce an untimed DES model which allows one to incorporate auxiliary data variables whose numeric information can be used to indicate buffer contents or priority. In this way, the occurrence of an event effects changes not only in the values of activity variables but also in the values of data variables. In addition to the feature of data variables, a two-valued function corresponding to each event acts as a guard to signal whether this respective event is qualified for exiting a state where the system resides.

By extending the untimed model, a framework is established for the control of a timed discrete-event system (TDES). Two key modelling features signify the differences between timed models and untimed models. The first is that a global digital clock is incorporated. In a vein similar to [1] and [4], the *tick* event is a newly-introduced event to indicate the advance of time and is allowed to be preempted from occurring. The second is that a default time bound interval is assigned to each non-tick event in a way similar to that in [3]. A default lower time bound is imposed to represent a delay in communication and a default upper time bound is imposed by physical constraints to indicate a deadline. All default time bound intervals imposed on non-tick events are measured with respect to the ticks of the clock. The eligibility of occurrence of non-tick events is influenced not only by the time bound intervals but also by the imposed enablement conditions.

The timed state-feedback control mechanism will be introduced for our design purpose. Finally, the merit of the model and timed state-feedback control design will be illustrated.

2 An Untimed Model

Definition 1 *An* untimed discrete-event model *G is a 5-tuple given by*

$$G := (\Sigma, \mathbf{X}, e, \delta, X_0),$$

where Σ is a finite set of event labels, $\mathbf{X}$ is a state set, e is a mapping, $e : \Sigma \to 2^{\mathbf{X}}$ where $2^{\mathbf{X}}$ denotes the set of functions : $\mathbf{X} \to \{0,1\}$, δ is a state transition function, $\delta : \mathbf{X} \times \Sigma \to \mathbf{X}$ and X_0 is the initial state.

$\Sigma := \Sigma_c \cup \Sigma_u$ where the disjoint subsets Σ_c and Σ_u consist of controllable events and uncontrollable events, respectively. Each state $X \in \mathbf{X}$ is described by a list of activity variables denoted by X_a and a list of data variables if they are present, which will be denote by X_d. With notational convenience, we will use e_σ instead of $e(\sigma)$. For each event $\sigma \in \Sigma$, $e_\sigma \in 2^{\mathbf{X}}$ and e_σ denotes the guard or the enablement condition for the event σ. Henceforth, we say that σ is *enabled* at a state X if $e_\sigma(X) = 1$, otherwise, σ is *disabled* at a state X. For each state $X \in \mathbf{X}$ the notation $\delta(X, \sigma)!$ denotes that $\delta(X, \sigma)$ is defined in G, while $\delta(X, \sigma)\neg!$ means that $\delta(X, \sigma)$ is not defined in G. $\delta(X, \sigma)!$ if and only if the event σ is enabled at X, i.e., $e_\sigma(X) = 1$.
The operation of G is usually displayed as a graph. An edge in a graph is pictured in Fig. 1, where

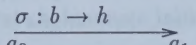


*Fig.*1

- σ is an event label of G;
- a_0 and a_1 are respectively the exit activity and the entrance activity;
- b represents a condition imposed on data variables that affects the enablement of σ;
- h is a list of arithmetic expressions for data variables affected by the occurrence of σ.

Furthermore, a tick on the edge shown as $\nrightarrow$ will be used to indicate that σ is a controllable event.

2.1 Semantics

A *state trajectory* $\mathcal{S}_G$ of G is defined to be an infinite sequence of states
$X_0 \xrightarrow{\sigma_0} X_1 \xrightarrow{\sigma_1} X_2 \xrightarrow{\sigma_2} \ldots$ where for each $i \geq 0, X_i \xrightarrow{\sigma_i} X_{i+1}$ is interpreted as X_i goes to X_{i+1} via the occurrence of the event σ_i. Each $\mathcal{S}_G$ of G must satisfy the following requirements:

- *Initialization* : X_0 is the initial state.
- *State Transition* : For each $i \geq 0$, consecutive states X_i and X_{i+1} satisfy that $\delta(X_i, \sigma)!$ and $X_{i+1} = \delta(X_i, \sigma)$.

The occurrence of events is by interleaving execution(i.e., no two distinct events can happen at the same time). When an event occurs, it occurs instantaneously. By the foregoing rule, the operation of an edge in Fig. 1 is described in the following. When an event σ occurs, the activity a_0 is taken to the activity a_1 and data variables are affected in accordance with the list h. For instance, let $h = [c_1 : c_1 - c_2]$. Thus, when σ occurs, c_1 is updated in accordance with $c_1 := (c_1 - c_2)$ while $c_2 := c_2$.

2.2 The Composition of DES

A large system is usually built up from smaller component systems, where each component is represented as a DES. To represent this complex system as a DES by bringing these components together, the notion of composition of DES is introduced. However, the rule is similar to that in the section 3.3. We omit it here.

3 A Timed Model

A DES G is extended to a TDES $\tilde{G}$ and is described as follows.

Definition 2 *Given a* timed model $\tilde{G}$ *as 6-tuple of the form*

$$\tilde{G} := (\tilde{\Sigma}, \tilde{\mathbf{X}}, \tilde{e}, s, \tilde{\delta}, \tilde{X}_0),$$

where $\tilde{\Sigma}$ *is a finite set of event labels with* $\tilde{\Sigma} := \Sigma \cup \{tick\}$, s *is a time bound interval mapping,* $s : \Sigma \to \mathbf{N} \times (\mathbf{N} \cup \{\infty\})$ *defined by* $s(\sigma) := [l_{\sigma,0}, u_{\sigma,0}]$ *where either* $l_{\sigma,0} \leq u_{\sigma,0}$ *if* $u_{\sigma,0} < \infty$ *or* $l_{\sigma,0} < u_{\sigma,0}$ *when* $u_{\sigma,0} = \infty$, $\tilde{\mathbf{X}}$ *is an augmented state set,* $\tilde{e}$ *is a mapping,* $\tilde{e} : \tilde{\Sigma} \to 2^{\tilde{\mathbf{X}}}$ *where* $2^{\tilde{\mathbf{X}}}$ *is a set of functions* $\tilde{\mathbf{X}} \to \{0, 1\}$, $\tilde{\delta}$ *is an augmented state transition function* $\tilde{\mathbf{X}} \times \tilde{\Sigma} \to \tilde{\mathbf{X}}$, $\tilde{X}_0$ *is the initial augmented state.*

The occurrence of a *tick* event indicates the passage of time. Due to the introduction of timing, a set of forcible events, Σ_f, is brought in naturally and $\Sigma_f \subseteq \Sigma$. A *tick* event is normally treated as an uncontrollable event. However, each event belonging to Σ_f is the one that can preempt the *tick* event. Hence, in this model, a *tick* event is thought of as controllable when a forcible event preempts the occurrence of a *tick* event. Thus, the status of a *tick* event is generally distinct from other non-tick events.
For each event $\sigma \in \Sigma$, $s(\sigma) := [l_{\sigma,0}, u_{\sigma,0}]$, where $l_{\sigma,0}$ is called the default lower time bound value and $u_{\sigma,0}$ is called the default upper time bound value corresponding to the event σ. The default upper time bound value for a controllable event is ∞. In the following, for each $\sigma \in \Sigma$, l_σ denotes the lower time bound and u_σ denotes the upper time bound variable.
For each augmented state $\tilde{X} \in \tilde{\mathbf{X}}$, $\tilde{X}$ is described by a list of activity variables and a list of data variables and is additionally described by a list of time bound intervals corresponding to each non-tick event $\sigma \in \Sigma$. When the global clock ticks, the time bound interval associated with each non-tick event may be affected. Therefore, the set of all possible time bound intervals for $\sigma \in \Sigma$ is

$$TM_\sigma := \begin{cases} \{[max(i + l_{\sigma,0} - u_{\sigma,0}, 0), i] | 0 \leq i \leq u_{\sigma,0}\} & \text{if } u_{\sigma,0} < \infty \\ \{[i, \infty] | 0 \leq i \leq l_{\sigma,0}\} \text{ otherwise} \end{cases}$$

As a result, the augmented state set $\tilde{\mathbf{X}} := \mathbf{X} \times \Pi\{TM_\sigma | \sigma \in \Sigma\}$.
For notational convenience, we will use $\tilde{e}_\sigma$, instead of $\tilde{e}(\sigma)$, to denote the guard or the enablement condition for the event $\sigma \in \tilde{\Sigma}$. For each $\sigma \in \tilde{\Sigma}$, we say that σ is *enabled* at an augmented state $\tilde{X}$ if $\tilde{e}_\sigma(\tilde{X}) = 1$, otherwise, σ is *disabled* at an augmented state $\tilde{X}$.
For each augmented state $\tilde{X} \in \tilde{\mathbf{X}}$ and each $\sigma \in \tilde{\Sigma}$, $\tilde{\delta}(\tilde{X}, \sigma)!$ denotes that σ is *eligible* at the augmented state $\tilde{X}$ in $\tilde{G}$, while $\tilde{\delta}(\tilde{X}, \sigma)\neg!$ means that σ is *not eligible* at the augmented state $\tilde{X}$ in $\tilde{G}$. For each $\tilde{X} \in \tilde{\mathbf{X}}$ and each $\alpha \in \Sigma$,

1. $\tilde{\delta}(\tilde{X}, \alpha)!$ if and only if $(\tilde{e}_\alpha \wedge (l_\alpha = 0))(\tilde{X}) = 1$, i.e., α is enabled at $\tilde{X}$ and the lower time bound associated with α in $\tilde{X}$ is 0,

2. $\tilde{\delta}(tick, \tilde{X})!$ if and only if $\tilde{e}_{tick}(\tilde{X}) = 1$.

It is straightforward to extend the picture of an edge in a graph of G to the picture of an edge in a graph of $\tilde{G}$ which is of the form in Fig. 2. $[l_{\sigma,0}, u_{\sigma,0}]$ is the default time bound interval for σ, and the other elements are described as before.

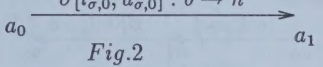


Fig.2

3.1 Semantics

The semantics of the timed models follow TTM's in [4]. An *augmented state trajectory* $\tilde{\mathcal{S}}_{\tilde{G}}$ of $\tilde{G}$ is an infinite sequence of augmented states of the form:
$\tilde{X}_0 \xrightarrow{\sigma_0} \tilde{X}_1 \xrightarrow{\sigma_1} \tilde{X}_2 \xrightarrow{\sigma_2} \ldots$. We interpret that for each $i \geq 0$, $\tilde{X}_i \xrightarrow{\sigma_i} \tilde{X}_{i+1}$ that $\tilde{X}_i$ goes to $\tilde{X}_{i+1}$ via the occurrence of σ_i where $\tilde{X}_i = (X_i, \{[l_\tau^i, u_\tau^i] | \tau \in \Sigma\})$ and $\tilde{X}_{i+1} = (X_{i+1}, \{[l_\tau^{i+1}, u_\tau^{i+1}] | \tau \in \Sigma\})$.
Any $\tilde{\mathcal{S}}_{\tilde{G}}$ must meet the following requirements:

- *Initialization* : $\tilde{X}_0$ is the initial augmented state.

- *Augmented State Transition* : For each $i \geq 0$, $\tilde{\delta}(\tilde{X}_i, \sigma_i)!$ and $\tilde{X}_{i+1} = \tilde{\delta}(\tilde{X}_i, \sigma_i)$. The state in $\tilde{X}_{i+1}$ is

$$X_{i+1} := \begin{cases} \delta(X_i, \sigma_i) \text{ if } \sigma_i \in \Sigma \\ X_i \text{ if } \sigma_i = tick \end{cases}$$

  The time bound interval associated with each event $\tau \in \Sigma$ in $\tilde{X}_{i+1}$ is derived from $\tilde{X}_i$ and will be shown in due course.

- *Ticking* : The ticking of a global digital clock cannott be stopped forever. In other words, a *tick* event must occur infinitely often. This eliminates the possibility that an infinite number of non-tick events occur consecutively without being interleaved by a tick event.

- *Time Bound Interval*: For each $i \geq 0$ and for each non-tick event α which is enabled at $\tilde{X}_i$, α can only occur within its associated time bound interval $[l_\alpha^i, u_\alpha^i]$ where u_α^i specifies the hard real-time value by which α must occur. If $u_\alpha^i = 0$, it would imply that α must be executed before a *tick* event unless α is preempted by the occurrence of other non-tick events. This enforces that a *tick* event can eligibly occur at $\tilde{X}_i$ if for every event $\alpha \in \Sigma$, $\tilde{e}_\alpha(\tilde{X}_i) = 1$ implies that $(u_\alpha > 0)(\tilde{X}_i) = 1$. This explains that $\tilde{e}_{tick} = \bigwedge_{\sigma \in \Sigma} \neg(\tilde{e}_\sigma \wedge (u_\sigma = 0))$. Hence, for each $i \geq 0$, $\tilde{\delta}(\tilde{X}_i, \sigma_i)!$, $\tilde{X}_{i+1} = \tilde{\delta}(\tilde{X}_i, \sigma_i)$ and the time bound interval corresponding to each $\tau \in \Sigma$ in $\tilde{X}_{i+1}$ is obtained as follows.

Case 1. $\sigma_i = tick$. Then

$$[l_\tau^{i+1}, u_\tau^{i+1}] = \begin{cases} [\max(0, l_\tau^i - 1), \max(0, u_\tau^i - 1)] & \text{if } \tilde{e}_\tau(\tilde{X}_i) = 1 \text{ and } u_\tau^i < \infty \\ [\max(0, l_\tau^i - 1), \infty] & \text{if } \tilde{e}_\tau(\tilde{X}_i) = 1 \text{ and } u_\tau^i = \infty \\ [l_{\tau,0}, u_{\tau,0}] & \text{otherwise} \end{cases}$$

Case 2. $\sigma_i \in \Sigma$. Then

$$[l_\tau^{i+1}, u_\tau^{i+1}] = \begin{cases} [l_\tau^i, u_\tau^i] & \text{if } \tau \neq \sigma_i,\ \tilde{e}_\tau(\tilde{X}_i) = 1 \text{ and } \tilde{e}_\tau(\tilde{X}_{i+1}) = 1 \\ [l_{\tau,0}, u_{\tau,0}] & \text{otherwise} \end{cases}$$

3.2 An Example

Given $\tilde{G}$ shown in Fig. 3 in which c is a data variable, the execution of any event in

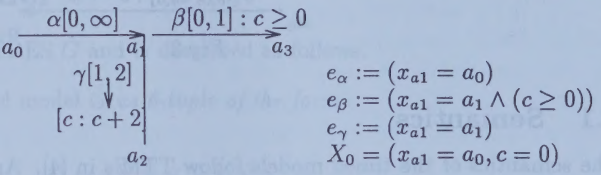


Fig.3

$\{\alpha, \beta, \gamma\}$ can be interpreted as follows. When α occurs, the new state has the activity a_1 and the value of data variable c remains unchanged. If γ occurs, the activity value of x_{a1} alters to a_2 and the data variable c is updated by the assignment $c+2$. β is enabled when the condition $x_{a1} = a_1 \wedge (c \geq 0)$ is satisfied. When β is executed, the new state is with activity a_3 and the value of data variable c remains unchanged. By the rule introduced in semantics, the augmented state trajectories of $\tilde{G}$ can be generated by traversing Fig. 4.

$$\begin{aligned}
\tilde{x}_0 &= (x_{a1} = a_0, c = 0, [0, \infty], [0, 1], [1, 2]) \\
\tilde{x}_1 &= (x_{a1} = a_1, c = 0, [0, \infty], [0, 1], [1, 2]) \\
\tilde{x}_2 &= (x_{a1} = a_2, c = 0, [0, \infty], [0, 1], [1, 2]) \\
\tilde{x}_3 &= (x_{a1} = a_2, c = 2, [0, \infty], [0, 0], [0, 1]) \\
\tilde{x}_4 &= (x_{a1} = a_3, c = 2, [0, \infty], [0, 1], [1, 2])
\end{aligned}$$

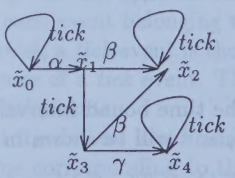


Fig.4

3.3 The Composition of TDES

For simplicity, we consider that given two TDES, M_1 and M_2, where

$$M_1 = (\tilde{\Sigma}_1, \tilde{\mathbf{X}}_1, \tilde{e}_1, s_1, \tilde{\delta}_1, \tilde{X}_{1o})$$

and

$$M_2 = (\tilde{\Sigma}_2, \tilde{\mathbf{X}}_2, \tilde{e}_2, s_2, \tilde{\delta}_2, \tilde{X}_{2o}).$$

Without loss of generality, suppose there is a shared data variable c and a shared event α in M_1 and M_2. A shared event for two components is an event synchronized between two components sharing the same event label; i.e., the execution of a shared event requires simultaneous participation of both components involved. Assume data variable assignments on the shared data variable c with respect to α and the initial value c_0 of the data variable c are consistent in M_1 and M_2.
The composite system of M_1 and M_2 is

$$M_1 \| M_2 = (\tilde{\Sigma}_3, \tilde{\mathbf{X}}_3, \tilde{e}_3, s_3, \tilde{\delta}_3, \tilde{X}_{3o}).$$

The event set $\tilde{\Sigma}_3 = \Sigma_1 \cup \Sigma_2 \cup \{tick\}$ and $\Sigma_1 \cup \Sigma_2$ is partitioned as disjoint subsets $\Sigma_{3,c}$ and $\Sigma_{3,u}$ where $\Sigma_{3,c} = \Sigma_{1,c} \cup \Sigma_{2,c}$ and $\Sigma_{3,u} = \Sigma_{1,u} \cup \Sigma_{2,u}$. Also, $\Sigma_{3,f} = \Sigma_{1,f} \cup \Sigma_{2,f}$

$s_3 : (\Sigma_1 \cup \Sigma_2) \to \mathbf{N} \times (\mathbf{N} \cup \{\infty\})$. For each event which is not the shared event α, its default time bound interval is as follows: for each $i \in \{1,2\}$ and for each $\sigma \in \Sigma_i$ with $\sigma \neq \alpha$, $s_3(\sigma) = s_i(\sigma)$. For the shared event α, its default time bound interval becomes $s_3(\alpha) = s_1(\alpha) \cap s_2(\alpha)$. In the following, assume $s_3(\alpha) = [l_{\alpha,0}, u_{\alpha,0}]$.

For each $\tilde{X}_3 \in \tilde{\mathbf{X}}_3$, $\tilde{X}_3$ is described by the list of activity variables $X_{3,a} = (X_{1,a}, X_{2,a})$ and the list of data variables $X_{3,d} = (X_{1,d} - \{c\}, X_{2,d} - \{c\}, c)$ and the list of time bound variables $\{[l_{3,\sigma}, u_{3,\sigma}] | \sigma \in (\Sigma_1 \cup \Sigma_2)\}$ where $X_{1,a}$ is the list of activity variables and $X_{1,d}$ is the list of data variables for describing a state $X_1 \in \mathbf{X}_1$ and $X_{2,a}$ is the list of activity variables and $X_{2,d}$ is the list of data variables for describing a state $X_2 \in \mathbf{X}_2$. The augmented state set $\tilde{\mathbf{X}}_3 := (\mathbf{X}_1 - \{c\}) \times (\mathbf{X}_2 - \{c\}) \times \{c\} \times \Pi\{TM_\sigma | \sigma \in (\Sigma_1 \cup \Sigma_2)\}$

$\tilde{e}_3$ is a mapping , $\tilde{e}_3 : \tilde{\Sigma}_3 \to 2^{\tilde{\mathbf{X}}_3}$. For each $\sigma \in (\tilde{\Sigma}_3 - \{\alpha, tick\})$, $\tilde{e}_{3,\sigma} = \tilde{e}_{i,\sigma}$, where $\sigma \in \Sigma_i, i \in \{1,2\}$. $\tilde{e}_{3,\alpha} = \tilde{e}_{1,\alpha} \wedge \tilde{e}_{2,\alpha}$ and $\tilde{e}_{3,tick} = \tilde{e}_{1,tick} \wedge \tilde{e}_{2,tick}$.

$\tilde{\delta}_3 : \tilde{\Sigma}_3 \times \tilde{\mathbf{X}}_3 \to \tilde{\mathbf{X}}_3$. For each $\sigma \in \tilde{\Sigma}_3$ and for each $\tilde{X}_3 \in \tilde{\mathbf{X}}_3$, $\tilde{\delta}_3(\tilde{X}_3, \sigma)!$ in M_3 if and only if $(\tilde{e}_{3,\sigma} \wedge (l_\sigma = 0))(\tilde{X}_3) = 1$.

$\tilde{X}_{3,0} = (X_{1,0} - \{c_0\}, X_{2,0} - \{c_0\}, c_0, \{[l_{\sigma,0}, u_{\sigma,0}] | \sigma \in (\Sigma_1 \cup \Sigma_2)\})$ is the initial augmented state.

These rules can be applied successively to building up a large system from any finite constituent processes.

4 Timed State-Feedback Controllers

Definition 3 *Given $\tilde{G}$, a* timed state feedback controller *for $\tilde{G}$ is a function $f : \tilde{\mathbf{X}} \to 2^{\tilde{\Sigma}}$ satisfying that for each $\tilde{X} \in \tilde{\mathbf{X}}$,*

$$\begin{array}{rl} \Sigma_u \cup \{tick\} \subseteq f(\tilde{X}) \subseteq \tilde{\Sigma} & \textit{if } tick \in f(\tilde{X}) \\ \Sigma_u \subseteq f(\tilde{X}) \subseteq \Sigma \textit{ and } (\Sigma_f \cap f(\tilde{X})) \neq \emptyset & \textit{otherwise} \end{array}$$

The controller receives the augmented state information output by $\tilde{G}$ and exercises control action determined by the current augmented state of $\tilde{G}$. For each $\tilde{X} \in \tilde{\mathbf{X}}$ and each $\sigma \in \tilde{\Sigma}$, we will write $f_\sigma(\tilde{X})$ instead of $f(\tilde{X})(\sigma)$. We may think that for each $\sigma \in \tilde{\Sigma}$, f_σ gives precisely the conditions under which events σ are enabled by the controller f.

4.1 Control Mechanisms

In the same way as [1], two distinct mechanisms employed explicitly by the control action of the controller f are introduced. One is enabling/disabling mechanism specifying for each non-tick controllable event whether it is enabled (permitted to occur) or it is disabled (disallowed to occur), and another is forcing mechanism which allows forcible events to preempt the *tick* event.

4.1.1 enabling/disabling mechanism

An event $\sigma \in \Sigma$ is disabled at $\tilde{X}$ by the controller f if $f_\sigma(\tilde{X}) = 0$, while σ is enabled at $\tilde{X}$ by the controller f when $f_\sigma(\tilde{X}) = 1$. For any event $\sigma \in \Sigma_c$, if we write $f_\sigma = false$, then for each $\tilde{X} \in \tilde{\mathbf{X}}$, $f_\sigma(\tilde{X}) = 0$. For each event $\sigma \in \Sigma_u$, $f_\sigma = true$ since for each $\tilde{X} \in \tilde{\mathbf{X}}$, $\Sigma_u \subseteq f(\tilde{X})$, i.e., uncontrollable non-tick events are always enabled.

4.1.2 forcing mechanism

The forcing mechanism is a built-in technology. $\neg f_{tick}$ characterizes those augmented states at which the forcing mechanism requires being invoked. When the forcing mechanism is employed at the current augmented state $\tilde{X}$, forcible events are endowed with authority to win the competition of occurrence over a tick event if a *tick* event and one or more forcible events are *eligible* at $\tilde{X}$. Under such circumstances, the forcing mechanism can be visualized by erasing the *tick* event from the set of events eligible at $\tilde{X}$.

5 Timed Closed-Loop Systems

By adjoining a timed state-feedback controller f to $\tilde{G}$, the controller f only affects the enablement condition of each event and augmented state transition function of $\tilde{G}$. Thus, the timed closed-loop model is defined as follows.

Definition 4 *Given $\tilde{G}$ and the timed state-feedback controller f for $\tilde{G}$, the timed closed-loop system $\tilde{G}^f$ is defined by*

$$\tilde{G}^f = (\tilde{\Sigma}, \tilde{\mathbf{X}}, \tilde{e}^f, s, \tilde{\delta}^f, \tilde{X}_0)$$

where $\tilde{\Sigma}, \tilde{\mathbf{X}}, s$ and $\tilde{X}_0$ remain the same as those in $\tilde{G}$, $\tilde{e}^f$ is a mapping $\tilde{\Sigma} \to 2^{\tilde{\mathbf{X}}}$, $\tilde{\delta}^f$ is an augmented state transition function: $\tilde{\mathbf{X}} \times \tilde{\Sigma} \to \tilde{\mathbf{X}}$.

For each $\sigma \in \tilde{\Sigma}$ and each $\tilde{X} \in \mathbf{X}$,

$$\tilde{e}^f_\sigma(\tilde{X}) := \begin{cases} 1 & \text{if } \tilde{e}_\sigma(\tilde{X}) = 1 \text{ and } f_\sigma(\tilde{X}) = 1 \\ 0 & \text{otherwise} \end{cases}$$

$\tilde{\delta}^f(\tilde{X}, \sigma)!$ denotes that the event $\sigma \in \tilde{\Sigma}$ is *eligible* at an augmented state $\tilde{X}$ in $\tilde{G}^f$ if $\tilde{\delta}(\tilde{X}, \sigma)!$ in $\tilde{G}$ and $f_\sigma(\tilde{X}) = 1$. On the other hand, $\tilde{\delta}(\tilde{X}, \sigma)\neg!$ denotes that σ is not eligible at $\tilde{X}$ in $\tilde{G}^f$.

Upon entering an augmented state $\tilde{X}$, f disables some events in Σ_c or preempts *tick* event such that the next possible event generated by $\tilde{G}^f$ is an event eligible at $\tilde{X}$. When an event $\sigma \in \tilde{\Sigma}$ occurs at an augmented state $\tilde{X}$, the system evolves to an augmented state $\tilde{\delta}^f(\tilde{X}, \sigma)$, i.e., $\tilde{\delta}(\tilde{X}, \sigma)$.

In line with RW theory in [5] and [6], we assume that the set of augmented state trajectories of $\tilde{G}^f$ is contained in the set of augmented state trajectories of $\tilde{G}$. In other words, each augmented state in $\tilde{G}^f$ is also an augmented state in $\tilde{G}$.

5.1 Discussion

Our timed model allows incorporation of data variables which don't exist in [1]. $\tilde{e}_{tick}$ and f_{tick} are introduced to express the condition that *tick* event should be preempted

from occurring in the operation of the plant and by the timed state-feedback controller, respectively. Although the graphical representation of our timed model is similar to TTM's introduced in [4], the key difference between our modelling formalism and TTM's is as follows. In TTM's, the enabling condition, time bound interval and transformation function for each non-tick event are grouped together as a transition. For this, a TTM's controller is the adequate controller form for control of a TTM's plant. Forcing mechanism employed in a TTM's controller design is related to the choice of both the enabling condition and the default upper time bound of a forcible event. This forcing design strategy has the result that the condition for this forcible event to preempt the occurrence of a *tick* event is the enabling condition associated with this forcible event. Our timed state-feedback controller setup allows the more flexible design choice that the conditions designated for a forcible event to preempt the occurrence of a tick event can be just subconditions of the enabling condition for this forcible event. These differences will be illustrated in the next section.

6 An Example

We consider a small manufacturing system which consists of one processing unit and two robots, say Robot1 and Robot2. The operations of Robot1 and Robot2 are as follows. Robot1 fetches raw material from an infinite input source and sends it to the processing unit. After that, Robot1 returns to the idle state. On the other hand, Robot2 retrieves the processed material from the processing unit and sends it to an infinite output sink. Subsequently, Robot2 returns to the idle state. Suppose the capacity of the processing unit is one. The activity variable for Robot1 is x_{a1} and the activity variable for Robot2 is x_{a2}. c is the data variable for the content of the processing unit. The graphical representation of this manufacturing system is shown in the following. The control objective to

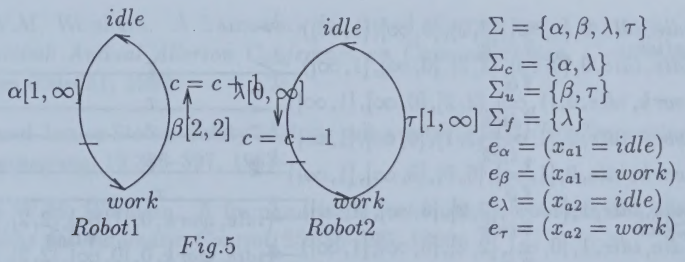


Fig.5

achieve is that the content of the processing unit should neither overflow nor underflow in the timed closed-loop system.

6.1 Solution

A TTM's controller is proposed as follows.
$\alpha[1,\infty] : \neg((x_{a1} = idle) \wedge (x_{a2} = work) \wedge (c = 1))$, $\beta[2,2] : true$,
$\lambda[0,1] : ((x_{a1} = work) \wedge (x_{a2} = idle) \wedge (c = 1))$, $\tau[1,\infty] : true$
λ is only enabled at any augmented state satisfying $(work, idle, 1)$ by this controller. Forcing mechanism is invoked at augmented states $\tilde{X}$ satisfying both $(work, idle, 1)$ and $(u_\lambda = 0)(\tilde{X}) = 0$ such that λ effectively erases the occurrence of the *tick*. The generated

controlled augmented state trajectories can be traversed by following the solid line in Fig. 6.

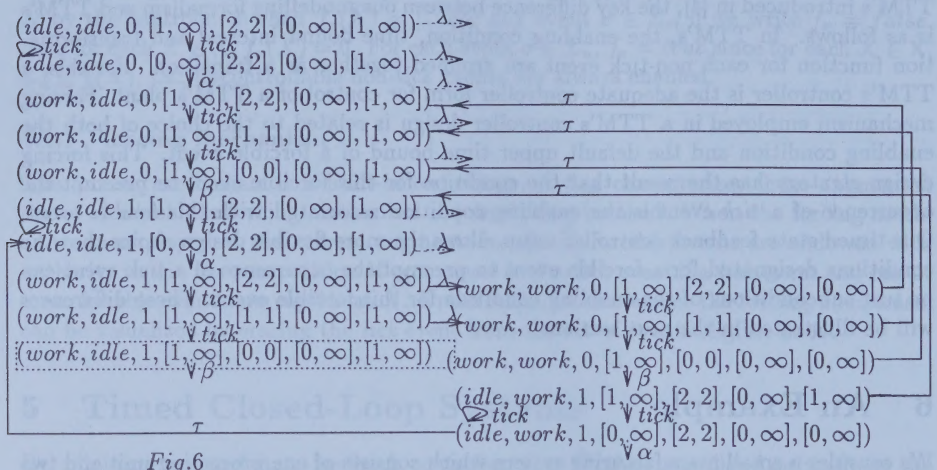

(idle, idle, 0, [1, ∞], [2, 2], [0, ∞], [1, ∞])
(idle, idle, 0, [0, ∞], [2, 2], [0, ∞], [1, ∞])
(work, idle, 0, [1, ∞], [2, 2], [0, ∞], [1, ∞])
(work, idle, 0, [1, ∞], [1, 1], [0, ∞], [1, ∞])
(work, idle, 0, [1, ∞], [0, 0], [0, ∞], [1, ∞])
(idle, idle, 1, [1, ∞], [2, 2], [0, ∞], [1, ∞])
(idle, idle, 1, [0, ∞], [2, 2], [0, ∞], [1, ∞])
(work, idle, 1, [1, ∞], [2, 2], [0, ∞], [1, ∞])
(work, idle, 1, [1, ∞], [1, 1], [0, ∞], [1, ∞])
(work, idle, 1, [1, ∞], [0, 0], [0, ∞], [1, ∞])
(work, work, 0, [1, ∞], [2, 2], [0, ∞], [0, ∞])
(work, work, 0, [1, ∞], [1, 1], [0, ∞], [0, ∞])
(work, work, 0, [1, ∞], [0, 0], [0, ∞], [0, ∞])
(idle, work, 1, [1, ∞], [2, 2], [0, ∞], [1, ∞])
(idle, work, 1, [0, ∞], [2, 2], [0, ∞], [0, ∞])
tick
α
β
λ
τ

*Fig.*6

An ad hoc timed state-feedback controller is proposed as follows.
$f_\alpha : \neg[(x_{a1}, x_{a2}, c) = (idle, work, 1)]$, $f_\beta : true$,
$f_\lambda : \neg[((x_{a1}, x_{a2}, c) = (idle, idle, 0)) \vee ((x_{a1}, x_{a2}, c) = (work, idle, 0))]$, $f_\tau : true$,
$f_{tick} : \neg[((x_{a1}, x_{a2}, c) = (work, idle, 1)) \wedge l_\lambda = 0 \wedge l_\beta = 1)]$
The forcing mechanism is invoked by the controller when any augmented state satisfies the condition $(((x_{a1}, x_{a2}, c) = (work, idle, 1)) \wedge l_\lambda = 0 \wedge l_\beta = 1)$. The *augmented state trajectories* in the timed closed-loop systems can be traversed by following the solid line in Fig. 7.

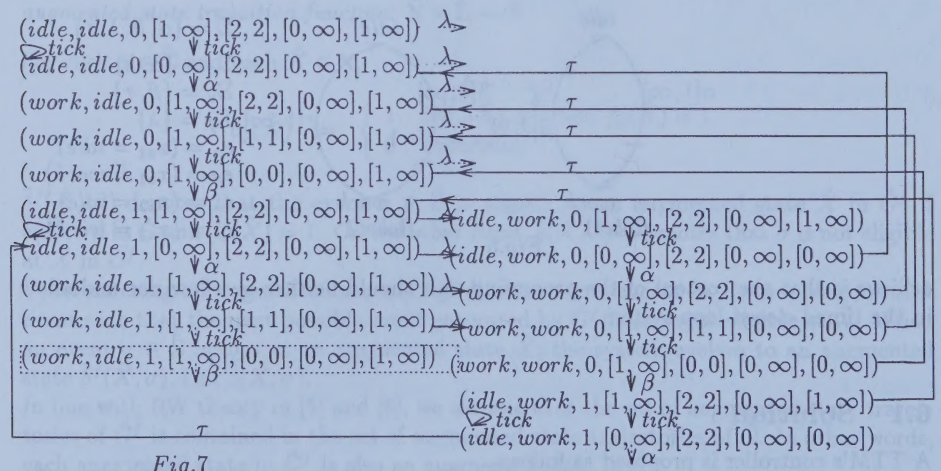


*Fig.*7

6.2 Discussion

1. Suppose the state-feedback controller only employs enabling/disabling mechanism. $(work, idle, 1, [1, \infty], [2, 2], [0, \infty], [1, \infty])$ and $(work, idle, 1, [1, \infty], [1, 1], [0, \infty], [1, \infty])$ in

Fig. 7 will not appear in the timed closed-loop system. This shows that the advantage of using both enabling/disabling and forcing mechanisms over merely employing enabling/disabling mechanism can achieve a richer closed-loop system behavior.
2. The proposed timed state-feedback controller for the example 6 shows that the set of controlled augmented state trajectories appearing in Fig.7 not only satisfies the specification but also is more permissive than that shown in Fig. 6 by using the TTM's controller.
3. The arithmetic property of the data variable is exploited in the modelling formalism and in the timed state-feedback controller design and representation.
4. A timed automata controller for this manufacturing system example can be obtained by using the synthesis approach in [1]. Though [1] solves the general TDES control problems, the computation and the representation of the resultant controller as a timed automata model are costly in general.

7 Conclusions

In this paper, we propose modelling formalisms for both the untimed model and the timed model. Examples are provided to illustrate the formal development.
The setup of the proposed timed state-feedback controller as shown in example 6 can concisely represent the control law to achieve the control purpose. This motivates us to focus on how to derive the timed state-feedback controller systematically and efficiently for a class of TDES control problems.

References

[1] B.A. Brandin and W.M. Wonham. Supervisory control of timed discrete-event system. *IEEE Transactions on Automatic Control*, 39:329–342, 1994.

[2] T.-J. Ho and W.M. Wonham. A framework for timed discrete event systems. In *Proceedings Thirtieth Annual Allerton Conference on Communication, Control and Computing*, pages 650–651, 1992.

[3] Nancy Leveson and Janice Stoley. Safety analysis using petri nets. *IEEE Transactions on Software Engineering*, 13:386–397, 1987.

[4] J.S. Ostroff and W.M. Wonham. A framework for real-time discrete-event control. *IEEE Transactions on Automatic Control*, 35:386–397, 1990.

[5] P.J. Ramadge and W.M. Wonham. Modular feedback logic for discrete event systems. *SIAM J. Control and Optimization*, 25:1202–1218, 1987.

[6] P.J. Ramadge and W.M. Wonham. Supervisory control of a class of discrete event processes. *SIAM J. Control and Optimization*, 25:206–229, 1987.

[7] W.M. Wonham. A control theory for discrete-event systems. In *Advanced Computing Concepts and Techniques in Control Engineering*, pages 129–169. Springer-Verlag NATO ASI Series, 1988.

A Supervisor Synthesis Algorithm for Stochastic Discrete Event Systems

Rick C.H. Chang and R.H. Kwong

Systems Control Group,
Department of Electrical and Computer Engineering
University of Toronto, Toronto, Ontario
Canada M5S 1A4

Abstract

We present a methodology for the design of stochastic discrete event control systems which better balances performance and logical objectives. A simple model of a stochastic discrete event system is to treat the underlying automaton as the generator of a Markov process. Based on stochastic analysis, we can design a supervisor that performs more efficiently because it permits some illegal states to be reached. However, this risk is maintained at an acceptable level and the necessity of a restart mechanism is duly accounted. The solution method is heuristic and thus only suboptimal results can be expected. We solve the synthesis problem for a transfer line example. Under the right conditions, it is shown that significant improvements in efficiency can be achieved by operating near system limits.

1 Introduction

Within the Ramadge-Wonham framework for control [10] [5] [7] [9] of a discrete event system (DES), there exists a systematic procedure for synthesizing a centralized supervisor. Both the plant and the supervisor are modeled as finite state automata. The plant is represented by an automaton:

$$G = (X, \Sigma, \xi, x_0),$$

where X is a finite set of discrete states and Σ is a finite set of events that can occur in the plant. Σ is also partitioned into controllable, Σ_c, and uncontrollable events, Σ_u. The mapping $\xi : X \times \Sigma \to X$ is a partial transition function that characterizes the plant dynamics. The closed behaviour $L(G)$ generated by G is a regular language. A complex plant is composed of component objects operating concurrently with each other. The plant DES can be constructed by taking the shuffle product of all component DES(s).

The event traces from the plant drives the state transitions of the supervisor automaton $S = (Q, \Sigma, \delta, q_0)$. The current supervisor state determines the subset of events enabled. For logical synthesis, the supervisor ensures that the controlled plant generates a specified language which is also realized by an automaton $E = (Y, \Sigma, \eta, y_0)$.

We use the following model for stochastic discrete event system (SDES):

$$\hat{S} = (S, \Omega)$$

where $\Omega : Q \times \Sigma \to \mathcal{R}$ is a function that associates each transition defined in DES S with the rate of an exponential random variable representing the transition time. We assign

an exponential rate to each event type. The resulting state transition structure generates a Markov chain. Based on this SDES model, we can include performance measures in supervisor synthesis for the system.

Consider the case of supervisor synthesis for the transfer line [2]. shown in figure 1. We assume that each buffer has capacity one. To ensure that the buffers do not overflow, the machines are made to wait (idle) at certain states of the supervisor. Suppose that the event in which a workpiece fails inspection (event 8) occurs very rarely. We can then keep machine 1 working and allow buffer one to overflow due to a workpiece being returned for reprocessing, provided that it does not occur too often. In fact, we want to design a controller such that the estimated risks are kept at acceptable levels. A restart mechanism is introduced to handle the buffer overflows and the cost of restarts is factored into the overall performance evaluation.

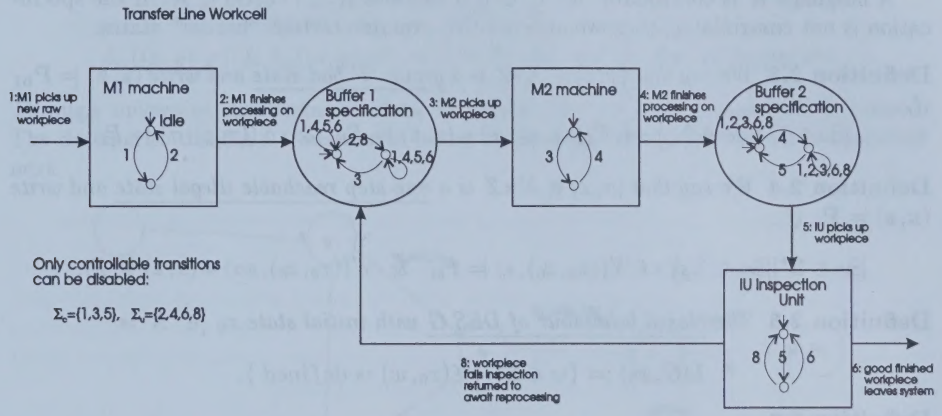


Figure 1: Schematic of a simple transfer line workcell.

In short, we want to design a supervisor that allows the system to operate efficiently and legally most of the time, with known, acceptable risks of illegality. We call this the "ε-synthesis" problem.

2 Definitions

For a string $s \in \Sigma^*$ that is generated by the plant, $s \in L(G)$, we want to augment the system state information $x = \xi(x_0, s)$ by also keeping track of the specification DES state that is reached.

We define a *monitor* DES(non-controlling supervisor)

$$S' = (X \times Z, \Sigma, \delta', (x_0, z_0))$$

where $Z = \{Y, V\}$ with the set V satisfying $V \cap Y = \emptyset$ and $z_0 = y_0$. The transitions which are driven by plant-generated events are defined for $\sigma \in \Sigma$, $(x, z) \in X \times Z$ as follows:

$$\delta'((x,z),\sigma) := \begin{cases} (\xi(x,\sigma), \eta(z,\sigma)) & \text{if } \xi(x,\sigma)! \ \& \ z \in Y \ \& \ \eta(z,\sigma)! \\ (\xi(x,\sigma), z') & \text{if } \xi(x,\sigma)! \ \& \ z' \in V \ \& \text{ not } \eta(z,\sigma)! \\ \text{undefined} & \text{otherwise}, \end{cases} \quad (1)$$

where ! denotes 'defined'.

If $z \in Y$, then we designate the system state (x, z) that is reached as *legal* since the generated sequence is accepted by the specification DES E. It is convenient to use the predicate notation for the following definitions.

Definition 2.1 *We say that* $(x, z) \in X \times Z$ *is a legal state and write* $(\mathbf{x}, \mathbf{z}) \models \mathbf{P_E}$ *if*

$$[\exists s \in \Sigma^*] \; s.t. \; s \in L(G) \cap L(E) \; \& \; \delta'((x_0, z_0), s) = (x, z).$$

Definition 2.2 *We say that* $(x, z) \in X \times Z$ *is an illegal state and write* $(\mathbf{x}, \mathbf{z}) \models \mathbf{P_I}$ *if*

$$[\exists s \in \Sigma^*] \; s.t. \; s \in L(G) \; \& \; s \notin L(E) \; \& \; \delta'((x_0, z_0), s) = (x, z), z \in V.$$

A language K is controllable w.r.t. G if it satisfies $\overline{K}\Sigma_u \cap L(G) \subseteq \overline{K}$. If the specification is not controllable, then we also need to examine certain "border" states.

Definition 2.3 *We say that* $(x, z) \in X \times Z$ *is a group '1' bad state and write* $(\mathbf{x}, \mathbf{z}) \models \mathbf{P_{b1}}$ *if*

$$[\exists s \in \Sigma^*][\exists \sigma \in \Sigma_u] \; s.t. \; \delta'((x_0, z_0), s) = (x, z) \models P_E \quad \& \quad \delta'((x, z), \sigma) \models P_I.$$

Definition 2.4 *We say that* $(x, z) \in X \times Z$ *is a one-step reachable illegal state and write* $(\mathbf{x}, \mathbf{z}) \models \mathbf{P_\varepsilon}$ *if*

$$[\exists s \in \Sigma^*][\exists \sigma \in \Sigma_u] \; s.t. \; \delta'((x_0, z_0), s) \models P_{b1} \quad \& \quad \delta'((x_0, z_0), s\sigma) = (x, z) \models P_I.$$

Definition 2.5 *The closed behaviour of DES G with initial state* $x_0 \in X$ *is*

$$L(G, x_0) := \{w \in \Sigma^* \mid \xi(x_0, w) \; is \; defined \; \}.$$

Definition 2.6

$$L(G, P_E \vee P_\varepsilon) := \{w \in L(G, x_0) \mid (\forall v \leq w) \; \delta'((x_0, z_0), v) \models P_E \vee P_\varepsilon\},$$

where $\leq$ *denotes 'prefix of'.*

In this investigation, we want to determine which transitions in S' to disable in order to obtain a supervisor $S = (Q \subseteq X \times Z, \Sigma, \delta, q_0 = (x_0, z_0))$ that enforces a language $K_\varepsilon \subset L(G, P_E \vee P_\varepsilon)$ that satisfies

1. **controllability condition**: $\overline{K_\varepsilon}\Sigma_u \cap L(G) \subseteq \overline{K_\varepsilon}$, and
2. **tolerance condition**: If for $s \in K_\varepsilon$, $q \in Q$, $\delta(q_0, s) = q \models P_\varepsilon$ then $p_q(\tau) \leq \varepsilon$ for $\varepsilon, \tau \geq 0$.

Here, $p_q(\tau)$ is the transient probability that the Markov chain generated by SDES $\hat{S} = (S, \Omega)$ is in state q at time $t = \tau$, τ is the time period over which we measure system performance, and ε is the upper tolerance limit for illegal behaviour to occur. Conditions 1 and 2 say that the supervisor which enforces K_ε must apply controls such that strings $s \in K_\varepsilon$ that reach states $q \models P_\varepsilon$ do so under the "frequency" constraint imposed by the probabilistic **tolerance condition**. For such a supervisor, we will treat states $q \models P_\varepsilon$ as absorbing states of the generated Markov chain. Later on we will add restart transitions at these states which effectively preempt any other transitions that take the supervised behaviour to other illegal states.

3 Supervisor Synthesis

We are interested in constructing a supervisor that will allow a system to run faster than when it is controlled by the supervisor (SUPCON) which enforces the supremal controllable sublanguage of the specification language. From [4], we know that such a supervisor must allow for a greater freedom of behaviour than SUPCON. Thus, our synthesis strategy is to carry out as few disablements as possible while having the enforced language respect the controllability and tolerance conditions. Instead of an exhaustive search, we present a heuristic scheme that yields a suboptimal solution.

The initial supervisor structure S_0 is constructed by adding all possible uncontrollable transitions from states $\models P_{b1}$ to states $\models P_\varepsilon$ to the DES $meet(G, E)$ defined below:

$$\begin{aligned} \text{meet} &: \text{DES} \times \text{DES} \longrightarrow \text{DES} \\ \text{meet}(G, E) &= (X \times Y, \Sigma, \delta_m, (x_0, y_0),), \\ \delta_m((x, y), \sigma)! \ \& \ \delta_m((x, y), \sigma) &:= (\xi(x, \sigma), \eta(y, \sigma)) \text{ iff } \xi(x, \sigma)! \ \& \ \eta(y, \sigma)! \end{aligned}$$

Through successive disablements, done in groups, the supervisor structure is pruned. The decision to disable a transition is based on its *proximity weight*, which is described next.

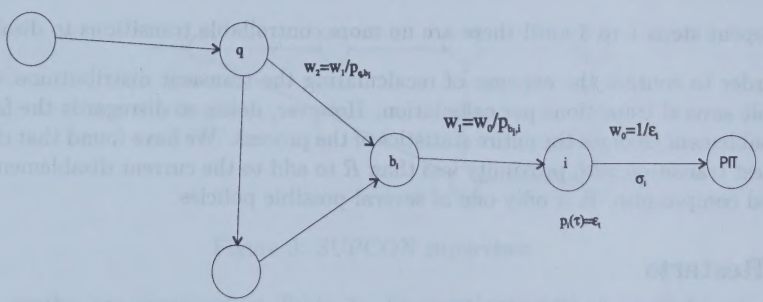


Figure 2: Example of Proximity Weight Assignments.

Proximity weights are assigned to transitions localized around states $\models P_\varepsilon$. We first construct imaginary transitions going from each absorbing illegal state to a single imaginary PIT state. From each state $i \models P_\varepsilon$, the proximity weight of the imaginary transition $w(i, \sigma_i, PIT)$ is assigned the value $1/\varepsilon_i$, where $\varepsilon_i = p_i(\tau)$ is obtained by calculating the transient distribution of SDES $\hat{S}_0 = (S_0, \Omega)$ at time $t = \tau$. Then, recursively, the upstream transition is assigned the current weight increased by the inverse of the upstream transition probability as the example shown in figure 2. Thus the larger the proximity weight is, the farther the transition is away from the PIT state. When a transition is assigned more than one proximity weight, we take the smallest value. Thus, we consider the higher likelihood path to the PIT state. Consider a value $R > 0$, termed the *range* and understood in terms of being of a certain proximity from the PIT state. The proximity test for a controllable transition (q, σ_c, q') is as follows:

> If $w(q, \sigma_c, q') > R$ then **enable** the transition, else it becomes a candidate for **disablement**.

For smaller ε, we need to set a larger R.

With this construction, we study the problem of reducing the probability accumulation at the absorbing states over finite τ. We want to steer the process away from these absorbing states by disabling high "traffic" paths. We start with those transitions closest to the PIT state because of our optimization objective.

3.1 Algorithm with Closest Proximity Disablement Policy

Starting with the initial supervisor structure,

1. For all states $\models P_\varepsilon$, calculate $p_i(\tau) = \varepsilon_i$ for each state. The transient probability distribution is calculated using the uniformization method as in [6].

   Eliminate any unreachable states and transitions.

2. For all absorbing states, recursively backward propagate the proximity weight $1/\varepsilon_i$ for each state i, stopping when the proximity exceeds R.

   According to the "Closest Proximity Disablement Policy", add the controllable transition with the lowest weight less than R to the current *disablement group.*

3. Disable the transitions in the current *disablement group.*

4. Repeat steps 1 to 3 until there are no more controllable transitions to disable.

In order to contain the expense of recalculating the transient distributuion we have to disable several transitions per calculation. However, doing so disregards the fact that each disablement changes the entire statistics of the process. We have found that choosing the closest transition with proximity less than R to add to the current disablement group is a good compromise. It is only one of several possible policies.

3.2 Restarts

We consider only non-disruptive restarts. This type of restarts does not interrupt any in-process plant activities.

A legal supervisor state can be labeled by (x, z) where $z \in Y$ is the corresponding state of the specification DES E. We will distinguish an illegal state with (x_i, z_i). The non-disruptive restart transition $((x_i, z_i), \sigma_r, (x_r, z_r))$ is defined by assigning the restarted state (x_r, z_r) as follows:

$$x_r = x_i \quad , \; z_r = z_{b1}$$

where (x_{b1}, z_{b1}) is the associated group '1' bad state from which (x_i, z_i) is reached. Note that the plant component remains unchanged because the restart action does not affect any ongoing plant processes. Also implied is the immediacy associated with the response time of these restarts.

4 Results for the Transfer Line

The data used for synthesis and evaluation is given in Table 1. For overall evaluation, each event is assigned a cost. This cost is additive eventwise in a string. Note the small rate associated with event '8'. This corresponds to low probability of a workpiece failing inspection. Also of significance are the high rates assigned to restarts to simulate the

fast responses required of restart transitions. The restarts amount to discarding the workpiece that caused the buffer to overflow, however this may actually be realized. The cost of the restart reflect the value of the discarded workpiece and in one situation, the workpiece has more added value because more processing has been done on it.

Parameter Data for Transfer Line			
Event	Event label	Rate per hr.	Event Cost
M1 picks up workpiece(w.p.)	1	60.	1.
M1 finish. process. w.p.	2	6.	5.
M2 picks up w.p.	3	60.	1.
M2 finish. proc. w.p.	4	6.	5.
IU picks up w.p.	5	60.	1.
Good w.p. leaves sys.	6	20.	3.
Failed w.p. recycled	8	0.01	3.
Discard w.p. in buff. 1	R1=9	180.	13.
Discard w.p. in buff. 2	R2=10	180.	21.

Table 1: Event parameters.

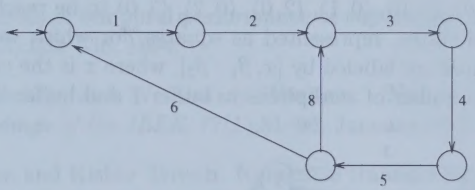


Figure 3: SUPCON supervisor.

The results are organized in Table 2. For synthesis with $\tau = 10$ *hours*, we can achieve $\varepsilon = 0.01$ by setting the range to $R = 500$. The performance of the ε-synthesized supervisor, shown in figure 5, is compared against that of the SUPCON supervisor, shown in figure 3, which enforces the supremal controllable sublanguage of the specification language. Over a 10 *hour* period, the expected number of event '6's is calculated. Each time this event occurs, a workpiece leaves the system and a profit of -43 is gained to offset the cost of production. We also tally the total cost of operating the supervised system over the same period to obtain the total net profit and the cost per unit produced. We gain a fifty per cent increase in productivity without any significant loss of profit margin.

Supervisor	E[#6's in $[0,\tau]$] Total Production	Total Net Profit	Total Oper. Cost	Cost per unit prod.
supcon	22.7284 (*)	-608.285	369.039	16.2369
ε-synthesis	34.1218 (*+50.13%)	-910.898	556.341	16.3045

Table 2: Results.

In figure 5, there are 32 states which are represented as circles. These states are labeled by pairs (x, y), where x and y are the current states of G and E, respectively.

G state label	(M1,M2,TU)	E state label	[Buf 1:Buf 2]
0	(I,I,I)	0	[0:0]
1	(W,I,I)	1	[1:0]
2	(I,W,I)	2	[0:1]
3	(I,I,W)	3	[1:1]
4	(W,W,I)		
5	(W,I,W)		
6	(I,W,W)		
7	(W,W,W)		

Table 3: State labels and correspondence.

We show the state labels of the plant and specification in figure 4 for reference. In Table 3, the corresponding physical states are also provided. The states which are marked by heavier-lined circles in figure 5 are called "producing states" because event '6' is possible from these states. Part of the structure is shown with dashed lines. Dashed-lined arcs are the disabled transitions and similar markings distinguishes states which have become unreachable as a result. In comparison with this supervisor, the SUPCON supervisor only allows states $(0,0)$, $(1,0)$, $(0,1)$, $(2,0)$, $(0,2)$, $(3,0)$ to be reached.

There are 6 illegal states, represented as squares, for which we must assign restart transitions. These states are labeled by $[x, \beta_1 : \beta_2]$, where x is the current state of G and β_1 and β_2 denote the number of workpieces in buffer 1 and buffer 2, respectively.

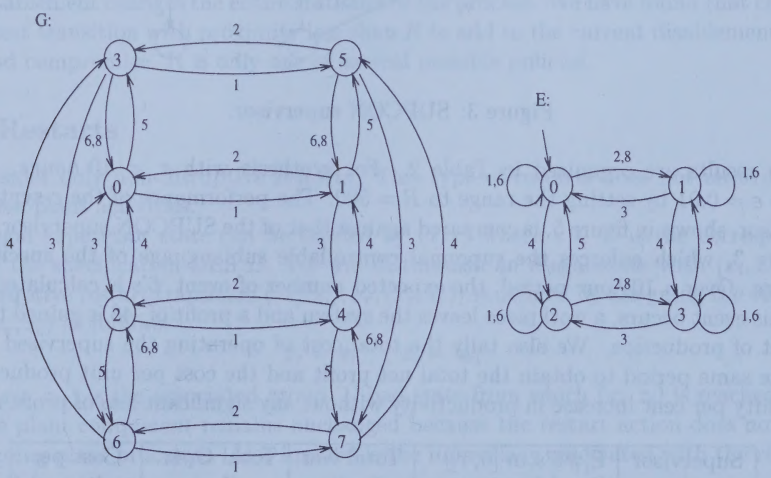


Figure 4: Plant and Specification.

Conclusion

We have presented an effective method for synthesizing a stochastic discrete event system supervisor which aims for a numerical performance objective in addition to the logical specification. We gain greater efficiency by using a supervisor that permits more freedom

of behaviour than the supervisor that enforces the supremal controllable sublanguage of the specification language. Consequently, some non-disastrous illegal states are allowed to be reached, though this risk is limited to an acceptable level. If conditions are such that much improvement can be had while the tolerance limit is kept small, then the heuristic algorithm gives good results.

References

[1] Rick C.H. Chang. Supervisor synthesis for stochastic discrete event systems. Master's thesis, University of Toronto, Graduate Department of Electrical & Computer Engineering, January 1996.

[2] R. H. Kwong and L. Zhu. *Performance Analysis and Control of Stochastic Discrete Event Systems.* Feedback Control, Nonlinear Systems and Complexity, B. A. Francis and A. R. Tannenbaum, eds., Lecture Notes in Control and Information Sciences, 202, Springer, 1995.

[3] Ying-Tat Leung and Rajan Suri. Performance evaluation of discrete manufacturing systems. *IEEE Control Systems Magazine*, pages 77–86, June 1990.

[4] Feng Ling. Analysis of temporal performance of supervised discrete event systems. *Automatica*, 30(3):533–536, 1994.

[5] Peter J. G. Ramadge and W. Murray Wonham. The control of descrete event systems. *Proceedings of the IEEE*, 77(1):81–98, January 1989.

[6] Andrew Reibman and Kishor Trivedi. Numerical transient analysis of markov models. *Comput. Opns Res.*, 15(1):19–36, 1988.

[7] Karen Gail Rudie. Software for the control of discrete event systems: A complexity study. Systems Control Group Report 8806, Systems Control Group, University of Toronto, September 1988.

[8] N. Viswanadham and Y. Narahari. *Performance Modeling of Automated Manufacturing Systems.* Prentice Hall, 1992.

[9] W. M. Wonham. *Notes on Control of Discrete-Event Systems.* Dept. of Electrical & Computer Engineering, Univ. of Toronto, revised 94.09.14 edition.

[10] W. M. Wonham and P. J. Ramadge. On the supremal controllable sublanguage of a given language. *SIAM J. Control and Optimization*, 25(3):637–659, May 1987.

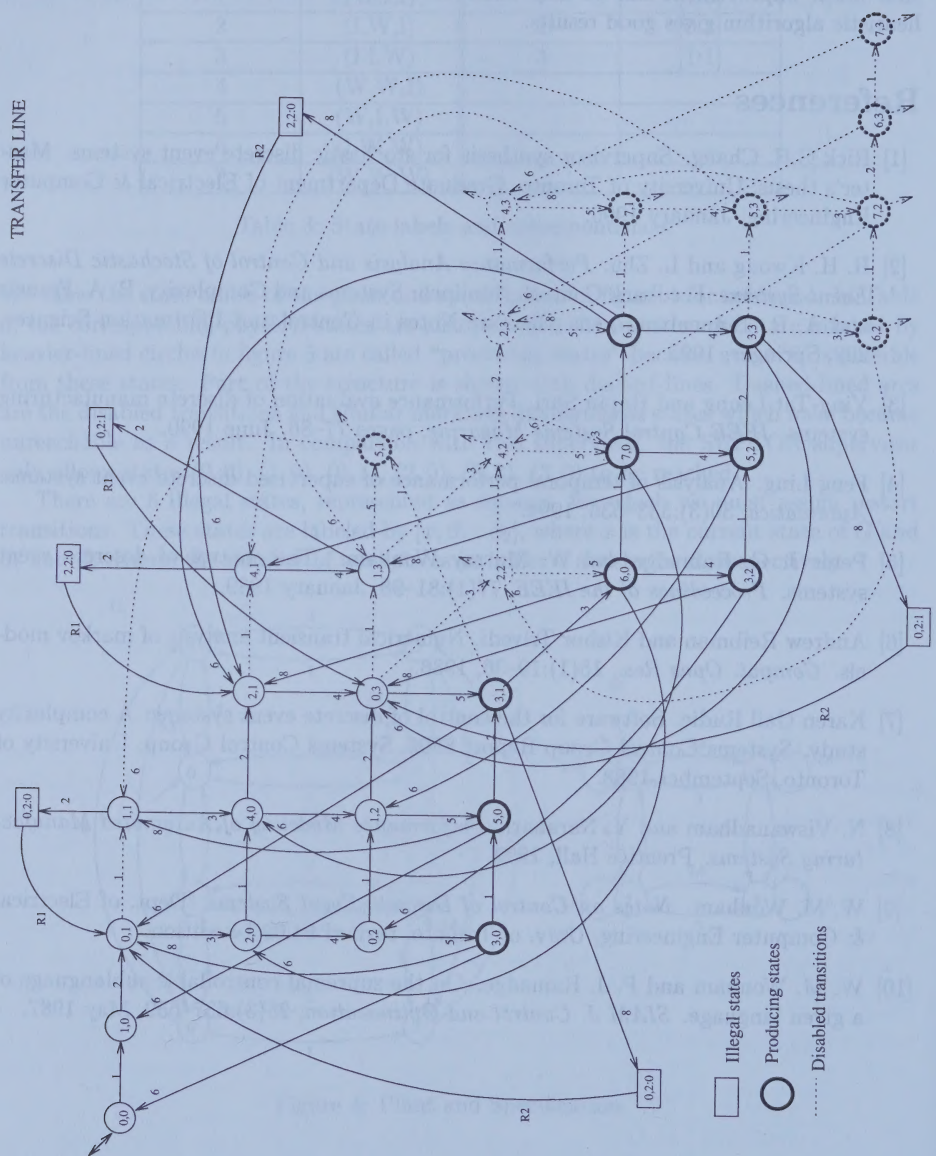


Figure 5: ε-synthesized supervisor.

Rule-Based Modeling of Hybrid Dynamic Systems Using Petri Nets

Moshe Gotesman
gotesman@cs.ttu.edu

Noe Lopez Benitez
nlb@cs.ttu.edu

Department of Computer Science
College of Engineering
Texas Tech University
Lubbock, Texas 79409
Voice: (806)742-1194
FAX: (806)742-3519

Abstract - A modeling technique for hybrid dynamic systems, using high-level Petri nets, is presented. A hybrid dynamic system is a continuous dynamic system in which discrete events occur. Modeling of hybrid dynamic systems is difficult since continuous dynamics and discrete event dynamics have entirely different mathematics. This paper presents a technique for converting a block diagram model of a continuous dynamic system into an equivalent Petri net. The Petri net captures both the continuous and discrete event phenomena in the system. This provides a single formalism for modeling hybrid dynamic systems.

Introduction

A hybrid dynamic system is a continuous dynamic system in which discrete events occur randomly. A continuous dynamic system is typically modeled by a system of differential equations [5]. The discrete events introduce non-continuous changes into the system that differential equations cannot handle.

Hybrid dynamic systems are found mainly in the manufacturing industry [4]. Autonomous manufacturing cells, chemical plants and power stations are typical examples of this type of systems [2], [11], [10]. Discrete events are also introduced into the system when it operates in a distributed computing environment. A distributed computing system is inherently a discrete event system. The communication network induces random delays in message transfer due to traffic load, communication protocol, etc. The nodes in the system are usually not synchronized. This generates random message arrival, job queues etc. These distributed computing phenomena affect the response of the dynamic system, typically degrading its performance and

stability [8].

Classical continuous dynamic system techniques cannot capture the discrete event phenomena. Similarly, discrete event modeling tools cannot handle continuous dynamics. As a result, modeling of hybrid dynamic systems is usually complex and inefficient. Typically, the engineer models the continuous aspects only, neglecting the discrete event impact on the dynamic behavior. The developer must use very robust design to compensate for the unknown effects of discrete events. This leads to an over conservative design.

The need for a single modeling tool that can capture both the continuous and the discrete event phenomena is clear. Modeling will be much easier and more accurate. A single systematic methodology will enable the user to perform a complete design in a unisied environment while capturing all the aspects of the system's behavior.

This paper presents a technique for modeling a hybrid dynamic system by means of Petri nets. The set of Petri nets used in this technique is a subclass of high-level Petri nets, and is referred to as Hybrid Dynamics Petri Nets (HDPN). The HDPN captures both continuous and discrete event phenomena. Thus, the hybrid dynamic system can be analyzed using conventional techniques and available tools for Petri nets. Both simulation and analysis techniques for Petri nets can be applied to the system. Thus, the power and flexibility of high-level Petri nets are available for modeling both "pure" continuous and hybrid dynamic systems.

Conversion of a Block Diagram to a Petri Net

The starting point of the modeling process is a specification of the continuous part of the hybrid dynamic system by a conventional block diagram. An example of the classical feedback loop system is shown in Figure 1.

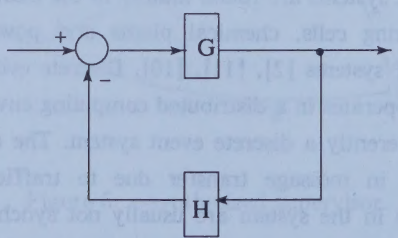


Figure 1: Example of a block diagram

The structure of the Petri net is derived from the structure of the block diagram. In [3], three techniques were developed for the conversion. One technique employs set theory and the two others are rule-based. In this paper, one rule-based technique is presented, named the "fast conversion technique".

Using this technique, the Petri Net structure can be derived by executing the following algorithm :

```
PROC BDS2CDPNS
    Replace each operational component (block, summation point, split point) with
    a transition.
    Plug a place on each arc.
ENDPROC
```

For example, Figure 2 demonstrates the application of the procedure to the example block diagram in Figure 1.

In the resulting Petri net, each transition (thin vertical bar) represents a block in the block diagram (summation points and split points are also considered "blocks" for this procedure). Places (circles) hold tokens (dots inside a place) that represent the signals traveling along the connections between the blocks in the block diagram. The Petri net at this stage can be viewed as a mapping of the structure of the dynamic systems for a single integration step.

The final phase in the modeling procedure is to add special elements to the Petri net to support dynamic features such as timing, signal computation and simulation execution. These features include:

A *master clock* transition *I*. Transition *I* is set to be a timed transition with a fixed delay, T_s, which corresponds to the integration time step. A timed transition fires only after the delay has elapsed. It is typically represented by a thick bar. All other transitions are immediate, that is they fire as soon as they are enabled. Every T_s seconds, transition *I* fires and transfers a token into the net. Since all other transitions are immediate, they cause the token to travel throughout the net until it arrives at place 1, where the computation cycle is complete. A new computation cycle will be triggered by the next firing of the master clock transition.

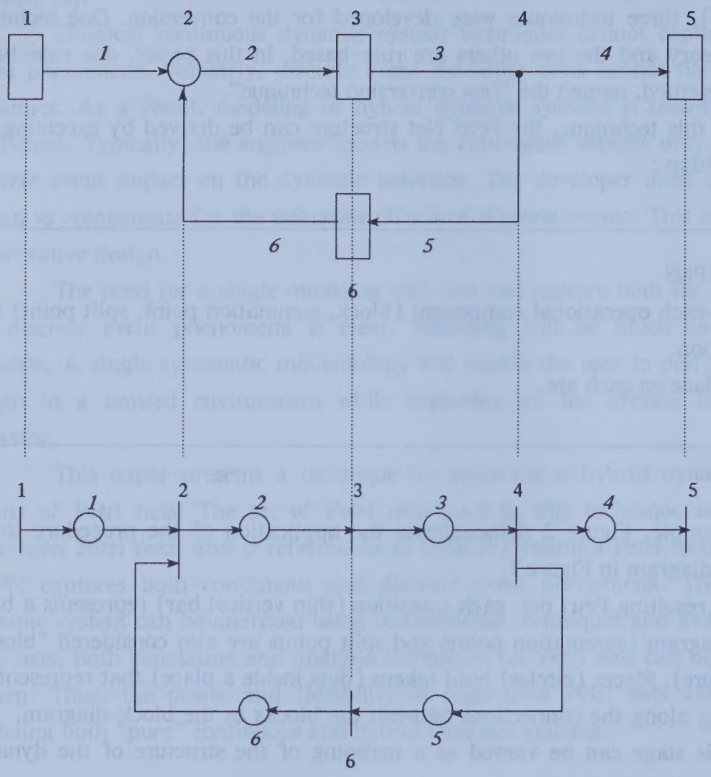


Figure 1: Quick conversion of the feedback loop system

2) *Action parameter tokens.* Each transition has an action associated with it. The action defines the signal processing required when the transition fires, which is equivalent to executing a transfer function associated with a block in the block diagram. Most actions require the specification of some parameters. For example, a transition associated with a transfer function needs to know the coefficients of the transfer function. For this purpose, each transition is given a dedicated parameter place, connected to it through a self loop. Each time the transition fires, the token in the parameter place provides the necessary parameters. This technique is an extension the technique suggested by [1].

The above additions account for the dynamics of the system and together with the structure comprise the complete specification of the HDPN. The resulting HDPN for the example in Figure 1 is shown in Figure 3.

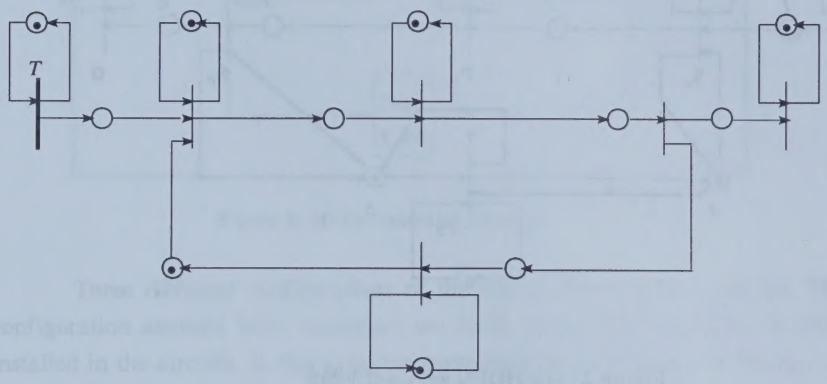


Figure 3: A complete HDPN

Networking Effects

When a dynamic/control system operates in a networked environment, this environment introduces additional effects into the system. Mostly, these effects are of a *discrete event* nature and affect the system by introducing delays into it. Modeling these delays by means of Petri nets raises issues such as *boundness* and *liveness*, which do not appear in Petri net models of non-networked systems. These issues must be addressed in a way that ensures that a single token travels in the net and triggers computation cycles at fixed integration step time intervals, regardless of the delay that the controller may introduce. The integration of delays due to networking effects is straight forward by assigning delays with the appropriate transitions.

Several solutions are suggested in [3] to ensure correct modeling of a dynamic system in a networked environment. A solution called "Self-Loops" is presented here.

In this approach, self loops are added at the appropriate transitions. Correct design of these loops will solve both the liveness and boundness problems. Figure 4 shows how this technique is applied to the example Petri net from Figure 3.

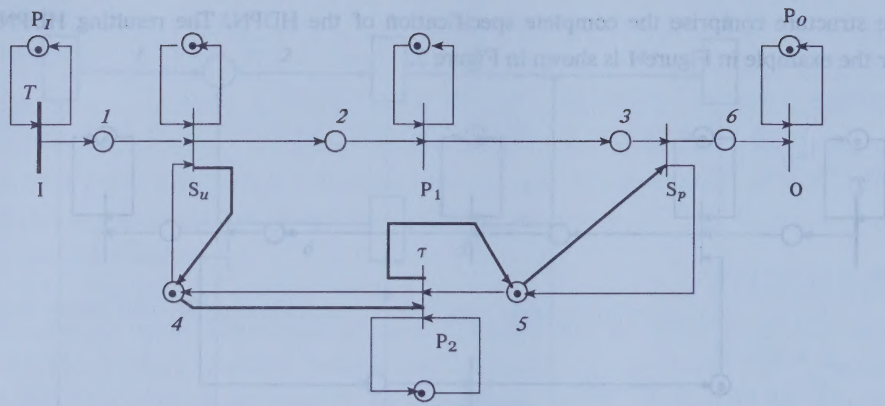


Figure 4: The HDPN with self loops

The additional self loops in the Petri net are shown in thick lines. In order to ensure liveness of the self loops, places *4* and *5* must contain each a single token in its initial marking. Each time transition S_u fires, the arc (S_u,*4*) "returns" a new token to place *4*. Thus, there is always a token in place *4* ensuring transition S_u is a live transition. Similarly, place *5* will always contain a token due to the loop arc (P_2,*5*). The loop arc (*5*,S_p) guarantees the replacement of the token in place *5* with a new token from place *3* when transition S_p fires. This prevents an accumulation of tokens in place *5* during the time transition P_2 is delayed. When transition P_2 fires, the "old" token in place *4* is removed through loop arc (*4*,P_2), and the token from place *4* is "transferred" to place *5*.

Example – Auto Pilot

This section brings an example that demonstrates the application of the principles, outlined in this paper to an auto pilot system. The BD for this problem is shown in Figure 5. The BD is a dual feedback system. The controlled variable is the aircraft's body angel, θ. θ_{ref} is the required angel. The system has two controllers, S_{amp} and S_{rg}, in the outer and inner loops, respectively.

The fast conversion technique was applied to generate the Petri net structure, shown in Figure 6. Next, the dynamics were incorporated into the net to producing the HDPN, shown in Figure 7. In this figure, the controller transitions are delayed, representing the effects of networking.

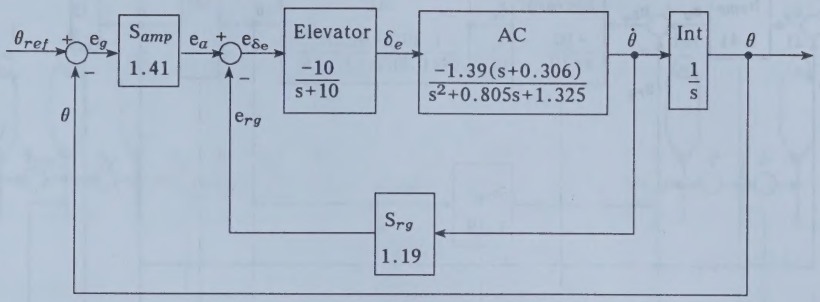


Figure 5: BD for autopilot example

Three different configurations of the above system were simulated. The first configuration assumes both controllers are local, that is, the auto pilot is physically installed in the aircraft. In this case, the controllers have no delay, rendering $\tau_1=\tau_2=0$. Thus, in this particular case, these transitions are immediate.

The second and third configurations assumed both controllers are remote. This situation may appear, for example, in unmanned aircraft, where part of the auto pilot is on the ground. Obviously, the need to transmit signals back and forth between the aircraft and the ground station introduces delays into the system. Also, it was assumed that a single CPU processes all the signals on a time share basis. In both remote cases, the CPU alternates its service between the two controllers in equal time slices. The difference between the to configurations is the policy that decides what is the control signal of a controller during the time slice when the CPU is servicing the other controller.

In one case, called *remote 1*, a controller that is not being serviced, generates a zero control signal. In the second approach, called *remote 2*, a controller that is not serviced by the CPU, freezes its output on the last value it had when it was last serviced.

The example was implemented in the Cabernet tool for modeling Petri nets ([7],[9]). The three configurations were simulated. The system's response is shown in Figure 8. The case of the local controller is shown with the solid line. Case Remote 1 is shown by the broken line and case Remote 2 by the broken-dot line.

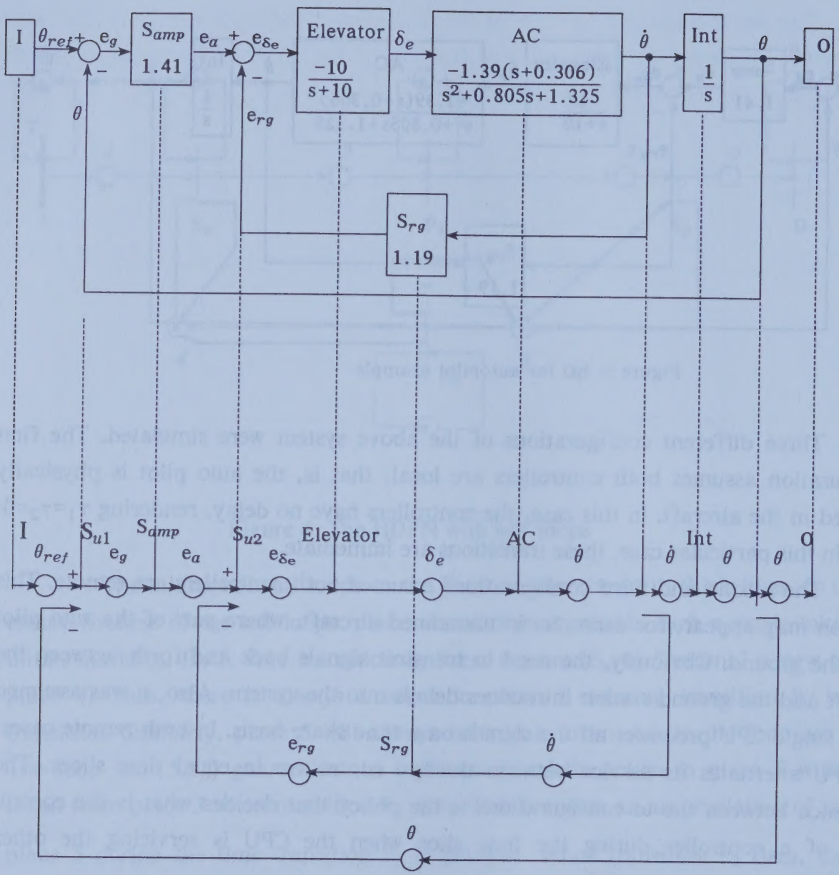


Figure 6: CDPNS for autopilot example

Conclusions

The paper demonstrates the capability of the HDPN to model hybrid dynamic systems. The HDPN can model the effects of networking in the dynamic system. The controller can be modeled as a control server which provides control service to multiple clients (plants) in parallel. In this context, the analysis of network protocols, queuing disciplines etc. is possible using a single formalism: Petri net theory.

An examples was presented, in which a HDPN was derived for a typical hybrid dynamic systems. The dynamic system was simulated by actually executing the "token game" in the HDPN. The response of the system as simulated by the HDPNs is presented.

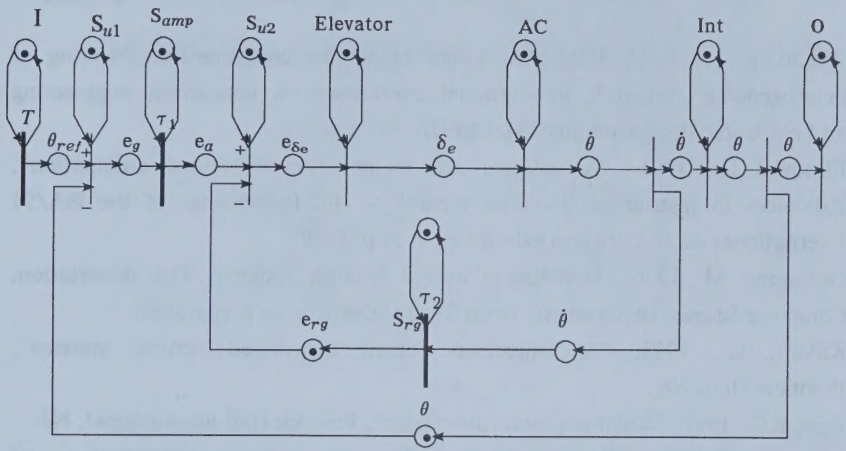


Figure 7: CDPN for autopilot example

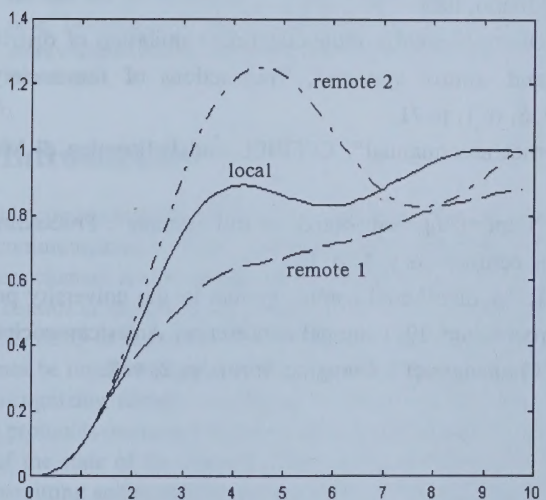


Figure 8: Simulation results for the autopilot

References

[1] Brielmann M. et al, 1994, "A design model for concurrent engineering of heterogeneous systems", International conference on concurrent engineering and electronic design automation, CEEDA.

[2] Elmqvist H., 1991, "A uniform architecture for distributed automation", Advances in instrumentation and control. v. 46. Proceedings of the ISA/91 international conference and exhibition, v. 2, p. 1599.

[3] Gotesman, M., 1996, "Modeling of hybrid dynamic systems", PhD dissertation, Computer Science Department, Texas Tech University, in preparation.

[4] Kalani, G., 1988, "Microprocessor based distributed control systems", Prentice-Hall, UK.

[5] Kuo, B.C., 1987, "Automatic control systems", Prentice Hall International, NJ.

[6] Mathworks, 1992, "Matlab, high performance numeric computation and visualization software" The Mathworks Inc.

[7] Pezze, M., 1994, "Caberenet: A customizable environment for the specification and analysis of real-time systems", Dipartmento di Electronica e Informazione, Politecnico di Milano, Milano, Italy.

[8] Ray, A. et al, 1988, "Discrete-event/continuous-time simulation of distributed data communication and control systems", Transactions of the society for computer simulation, v. 5, n. 1, p. 71.

[9] Silva, S., 1994, "Cabernet user manual", CEFRIEL and Politecnico di Milano, Milano, Italy.

[10] Thakkar, J.K., 1991, "Expertizing distributed control systems", Proceedings of the industrial computing conference. v. 1, p. 123.

[11] Waggoner, R. C., 1991, "A distributed control system in the university process control laboratory", Proceedings 1991 annual conference, American society for engineering education. Challenges of a changing world., p. 2, v. 2.

Error Control and Energy Consumption in Wireless Communications Channels

Michele Zorzi and Ramesh R. Rao

Center for Wireless Communications
University of California, San Diego
9500 Gilman Dr. – La Jolla, CA 92093-0407
fax: (619) 534-2486 – E-mail: zorzi@ece.ucsd.edu, rrao@ucsd.edu

Abstract

In this paper, the problem of error control over an energy constrained wireless channel is studied. Errors on both the forward and the feedback channels are modeled as Markov. Classic ARQ strategies are found to lead to a considerable waste of energy, due to the large number of transmissions. An adaptive scheme, which slows down the transmission rate when the channel is impaired, is shown to save energy, without a significant loss in throughput. Some references on the modelling of battery cells are discussed to highlight the fact that battery charge capacity is strongly influenced by the available "relaxation time" between current pulses. Motivated by this, an analytical approach that can track more complex models for power sources, including dynamic charge recovery, is formally established.

1 Introduction

Portable communication devices must inherently rely on batteries with limited energy to conduct communications. At the same time, due to fading and other propagation impairments, the wireless channel is error prone. In this context, there is a need to understand how to perform error control in an energy constrained environment. Well known error control schemes such as FEC, ARQ or Hybrid ARQ have all been used to overcome channel outages. Can these schemes be improved upon in the context of energy constraints?

Recognizing energy constraints creates new opportunities for protocol design. Classic ARQ protocols overcome errors by retransmitting the erroneously transmitted packet regardless of the state of the channel. Thus, even when the channel is bad, these protocols keep retransmitting and expending energy since such action does not cost anything. In fact, as soon as the channel becomes good again, packets are received correctly with minimal delay. In contrast, in the energy constrained environment, repeated retransmission when the channel is bad depletes the limited energy available. Subsequently, when the channel becomes good again the depletion in energy may make transmissions more prone to errors resulting in potentially more retransmissions, leading to a catastrophic loss of energy. Clearly, such persistence may not be desirable when operating under energy constarints. This raises the possibility that upon

detecting transmission failures, the sender refrains from transmitting data until the channel has recovered. Thus, some of the useless transmissions performed during an OFF period can be avoided.

As a first step towards addressing such questions systematically, we propose and analyze an ARQ protocol in which, when the channel conditions deteriorate, the transmitter enters a *probing mode*, and sends a short probing packet every so often. When channel conditions improve, the transmitter switches back to the *normal mode* and restarts transmission from the point it was interrupted. The probe packet can be thought of as a packet with a header but either no or little payload. The value of this approach of course depends on how rapidly fading occurs relative to the round trip delay time on the link. We find that accepting a moderate throughput reduction makes it possible to significantly reduce the energy consumption of the system, relative to classic ARQ schemes. Besides reducing energy consumption, minimizing the number of transmissions per correctly received packet may be desirable in multiple access environments as a form of interference reduction.

Next, as a step towards addressing the modelling of battery sources, we first assume that the battery charge is a step function, providing a constant output as long as there is some charge and then dropping to zero when the battery is discharged [1]. With this assumption, we find that a steady state analysis of the scheme described above can be done without having to track detailed state information. On the other hand, when the available battery charge profile displays a more complex functional form, possibly including charge recovery, it becomes necessary to expand the model and track a larger set of state variables. This detailed approach [1], outlined in Section 4, is powerful, not only because it provides complete statistical information but also because it allows *all quantities* to depend on the energy state, u.

The paper is organized as follows. In Section 2 the protocol and the channel model are described. The Markov analysis of the adaptive scheme is presented in Section 3. An extended analytical model which takes into account state-dependent behavior is discussed in Section 4.

2 Protocol description and channel model

Consider an ARQ protocol which operates as follows. The transmitter sends one block per slot, as long as it senses that both channels are ON: this can be easily tracked by checking that ACKs are properly received. As soon as the transmitter detects that either channel is in error, it enters a *probing mode*, in which it sends a probing packet every t slots. This mode of operation is continued until an ACK is received, acknowledging the good status of both channels: this event makes the transmitter switch back to the *normal mode* and restart transmission from the point it was interrupted. The probe packet can be thought of as a packet with a header but either no or little payload.

In summary, the receiver responds with an ACK every time it detects a valid transmission, and with a NAK every time it fails to do so. The transmitter acts according to the following rules:

- in the normal mode (N), it sends packets and receives the corresponding feedback after m slots from the beginning of the transmission (round-trip);
- it switches from N to the probing mode (P) upon receiving a NAK or a garbled feedback;
- while in P, it transmits a probing packet every $t \geq m$ slots;

- it switches from P to N as soon as it gets an ACK to a probing packet.

The patterns of packet errors on the forward and reverse channels are assumed to follow two independent first-order Markov models. This is a good model for packet transmission over a wireless fading channel, as discussed in [2]. The two transition matrices are $\mathrm{M}_F(x) = \mathrm{M}_F(1)^x$ and $\mathrm{M}_B(x) = \mathrm{M}_B(1)^x$, with

$$\mathrm{M}_F(x) = \begin{pmatrix} p(x) & q(x) \\ r(x) & s(x) \end{pmatrix}, \quad \mathrm{M}_B(x) = \begin{pmatrix} a(x) & b(x) \\ c(x) & d(x) \end{pmatrix}, \tag{1}$$

$$\mathrm{M}_F(1) = \begin{pmatrix} p & q \\ r & s \end{pmatrix}, \quad \mathrm{M}_B(1) = \begin{pmatrix} a & b \\ c & d \end{pmatrix}, \tag{2}$$

where $p(x) = 1 - q(x)$ ($r(x) = 1 - s(x)$) is the probability that the forward block i is successful given that the forward block $i - x$ was successful (unsuccessful), and similarly for the entries of $\mathrm{M}_B(x)$, with reference to the backward channel. The average block error rate is found as $\varepsilon = q/(r+q)$. The Markov parameters can be directly related to the physical channel characterization, such as the fading margin and the Doppler frequency [2].

We assume, in the following, that the packet length is a constant, equal to one time unit (slot). This assumption is consistent with the current trend towards ATM-compatible transport modes, where fixed size packets are envisioned. The round-trip delay (accounting for propagation and processing times) is $m - 1$ slots from the end of a transmission to the reception and decoding of the corresponding feedback information, so that if a packet transmitted in slot i is negatively acknowledged, it will be retransmitted in slot $i + m$. We assume that positive (ACK) and negative (NAK) acknowledgements can never be confused with each other, i.e., the effect of backward errors is to map the ACK and NAK symbols to an Erasure symbol. Also, we assume that each ACK/NAK carries the necessary information to identify all past correctly received packets.

3 Protocol Evolution

The evolution of the protocol depends on the status of the channel, described by four states, namely, 1 (E/C), 2 (C/C), 3 (E/E) and 4 (C/E), where the notation in brackets stands for the outcomes of the forward and backward channels, respectively, with E denoting erroneous and C denoting correct transmissions. Each transition occurs right after the packet transmission.

The time instants at which the two channels are sampled are separated by $m-1$ slots, corresponding to the round-trip delay. Therefore, each state of the channel tracks the transmission of a packet and the reception of the corresponding ACK/NAK. This is done for purpose of analysis, as in [3]. The time displacement is of course taken into account when describing the protocol rules and the transition structure.

3.1 Transition structure

In the regular mode of operation (N), an ACK is received in every slot, and the system keeps making transitions from state 2 back to state 2. These transitions occur at a rate of one per slot. Define the *time of stay (TOS)* in a state as the number of slots the system stays inactive in that state. The TOS in state 2 is always one slot, and the probabilities of the transitions exiting state

2 are found from the one-step matrices $\mathbf{M}_F(1)$ and $\mathbf{M}_B(1)$. For example, the probability of going from state 2 to itself is ap (i.e., $P[C|C]$ on both channels, using the one-step matrices). The probability of going from state 2 to state 1 is aq (i.e., $P[E|C]$ on the forward channel and $P[C|C]$ on the feedback channel), and so on.

On the other hand, upon leaving state 2, the system enters the probing mode, where a probe transmission and the corresponding feedback occur every t slots. Therefore, the TOS in states 1, 3 and 4 is t slots, since a transition from those states occurs right after the transmission of the next probe packet. The transition probabilities can be derived as before, except for the fact that now the t-step matrices, $\mathbf{M}_F(t)$ and $\mathbf{M}_B(t)$, are to be considered.

In our analysis, the pair of outcomes on the two channels are tracked together. In the actual system a transition from states 1, 3 or 4 to state 2 certainly takes place when C/C occurs, but only m slots later will the transmitter be aware that a transition to mode N has occurred, when it actually receives the ACK. After that, it starts transmitting one block per slot. Therefore, the behavior in state 2 depends on where the transition came from. In order for the model to be Markovian, state 2 must therefore be split into two states, which we call 2 and 2', where state 2' is entered when a C/C occurs given that the system was in states 1, 3 or 4. Due to the time it takes to the transmitter to detect the resumption of the N mode, the TOS in 2' is therefore m slots. Also, state 2' is entered from states 1, 3 and 4, whereas state 2 can be entered only from 2' or itself.

Based on the above discussion, the transition matrix can be found, and is given by

$$\mathbf{P} = \begin{pmatrix} a(t)s(t) & 0 & a(t)r(t) & b(t)s(t) & b(t)r(t) \\ aq & ap & 0 & bq & bp \\ a(m)q(m) & a(m)p(m) & 0 & b(m)q(m) & b(m)p(m) \\ c(t)s(t) & 0 & c(t)r(t) & d(t)s(t) & d(t)r(t) \\ c(t)q(t) & 0 & c(t)p(t) & d(t)q(t) & d(t)p(t) \end{pmatrix}, \tag{3}$$

with states ordered as 1, 2, 2', 3, 4; the TOS for each state is summarized by the vector

$$\begin{pmatrix} t & 1 & m & t & t \end{pmatrix}. \tag{4}$$

3.2 Throughput performance

Let i be a state, arbitrarily chosen as a reference, and let π_i denote its steady-state probability. The vector π is the steady-state distribution of the Markov chain, i.e., the solution of the equation

$$\pi = \pi\mathbf{P}, \quad \sum_{i\in\Omega} \pi_i = 1, \tag{5}$$

where $\mathbf{P}$ is the transition matrix, whose elements $P_{ij} = P[j|i]$ are the transition probabilities, and $\Omega = \{1, 2, 2', 3, 4\}$. Also, let D_{ij} be the time delay associated to transition from state i to state j and define the mean waiting time in state i as

$$D_i = \sum_{j\in\Omega} P_{ij} D_{ij}. \tag{6}$$

In steady state, the mean recurrence time of state i, $\overline{D}_{ii}$, defined as the average time between two consecutive entrances into state i, is given by (see [4], p. 641):

$$\overline{D}_{ii} = \frac{\sum_{\ell\in\Omega} \pi_\ell D_\ell}{\pi_i}. \tag{7}$$

Equation (7) holds for a generic reward function as well. If R_{ij} is the reward (e.g., the number of successful transmissions, i.e., of packets which have been correctly received *and* acknowledged) associated to the transition from state i to state j, we have, as in (6) and (7),

$$R_i = \sum_{j \in \Omega} P_{ij} R_{ij}, \tag{8}$$

and

$$\overline{R}_{ii} = \frac{\sum_{\ell \in \Omega} \pi_\ell R_\ell}{\pi_i}. \tag{9}$$

From a fundamental theorem of renewal reward processes, we have that [5]

$$\lim_{\tau \to \infty} \frac{R(\tau)}{\tau} = \frac{\overline{R}_{ii}}{\overline{D}_{ii}} = \frac{\sum_{\ell \in \Omega} \pi_\ell R_\ell}{\sum_{\ell \in \Omega} \pi_\ell D_\ell}, \tag{10}$$

where $R(\tau)$ is the number of rewards (i.e., of correct receptions) at time τ, and $\overline{R}_{ii}$ is the average number of rewards during a cycle. A cycle is defined as the time between two consecutive passages for the reference state, i (renewals). Note that in (10) the probability π_i cancels out, and the result is independent of the choice of the reference state, as expected [4]. Note also that, if the rewards are the successful transmissions, Eq. (10) gives the steady-state throughput efficiency.

In the present context, the transitions from 2, 2' to 2 involve a single reward, whereas all others have none. The reward matrix is given by

$$\mathbf{R} = \begin{pmatrix} 0 & - & 0 & 0 & 0 \\ 0 & 1 & - & 0 & 0 \\ 0 & 1 & - & 0 & 0 \\ 0 & - & 0 & 0 & 0 \\ 0 & - & 0 & 0 & 0 \end{pmatrix}, \tag{11}$$

and from (8) we obtain

$$R_2 = ap, \;\; R_{2'} = a(m)p(m), \;\; R_1 = R_3 = R_4 = 0. \tag{12}$$

Finally, the delays D_ℓ's are just the components of (4). Therefore, the throughput performance of the protocol can be easily computed from (10), after the associated Markov chain has been solved.

3.3 Number of transmissions

The above approach can be generalized: if $X(\tau)$ and $Y(\tau)$ are two reward functions, from the renewal reward process theory, we have [5]:

$$\lim_{\tau \to \infty} \frac{X(\tau)}{Y(\tau)} = \frac{E[X]}{E[Y]} = \frac{\sum_{\ell \in \Omega} \pi_\ell X_\ell}{\sum_{\ell \in \Omega} \pi_\ell Y_\ell}. \tag{13}$$

This equation will be employed in the following to compute the average number of transmissions per successful transmission, with $X = T$ (total number of transmissions) and $Y = R$ (number of successful transmissions).

A transition from states 2 and 2' to 2 involves one block transmission, transitions from states 2, 2' to states in P involve transmission of m blocks, whereas all outgoing transitions

from states 1, 3 and 4 involve a probing packet transmission. The matrix of the block transmissions associated with the transitions is therefore

$$\mathbf{T}^{(b)} = \begin{pmatrix} 0 & - & 0 & 0 & 0 \\ m & 1 & - & m & m \\ m & 1 & - & m & m \\ 0 & - & 0 & 0 & 0 \\ 0 & - & 0 & 0 & 0 \end{pmatrix}, \tag{14}$$

and that of the probing packet transmissions is

$$\mathbf{T}^{(p)} = \begin{pmatrix} 1 & - & 1 & 1 & 1 \\ 0 & 0 & - & 0 & 0 \\ 0 & 0 & - & 0 & 0 \\ 1 & - & 1 & 1 & 1 \\ 1 & - & 1 & 1 & 1 \end{pmatrix}. \tag{15}$$

If the energy associated with a block transmission is normalized to unity and that of a probing packet is denoted by $w \leq 1$, the matrix of the transmitted energy (measured in equivalent block transmissions) can be written as

$$\mathbf{T} = \mathbf{T}^{(b)} + w\mathbf{T}^{(p)} = \begin{pmatrix} w & - & w & w & w \\ m & 1 & - & m & m \\ m & 1 & - & m & m \\ w & - & w & w & w \\ w & - & w & w & w \end{pmatrix}. \tag{16}$$

The average number of equivalent transmissions per successful transmission is therefore given by

$$N = \frac{\sum_{\ell \in \Omega} \pi_\ell T_\ell}{\sum_{\ell \in \Omega} \pi_\ell R_\ell}. \tag{17}$$

3.4 Results

Based on the above analysis, we obtained some numerical results for the throughput and energy consumption performance of the proposed protocol, and we compared them to the classic Go-Back-N ARQ protocol. As expected, we found that our adaptive strategy has a smaller throughput than the classic protocol, but it exhibits improved energy consumption performance. Minimizing unsuccessful transmissions reduces the throughput, since some good slots are left empty. On the other hand, to maximize the throughput one has to always attempt transmission possibly wasting energy [1].

As an example, Figs. 1 and 2 highlight the trade-off between throughput and energy consumption. The average number of equivalent block transmissions, N (Eq. (17)), is plotted vs. the throughput η, and the curves are parametrized by the probing interval, t. (Different curves are drawn for various channel statistics, corresponding to various values of the fading margin [1, 2].) The performance of classic ARQ is represented by a single point. It is clear that the gain in transmission energy saving can be very significant, especially for harsh channel conditions. Also, it can be seen that for $m = 1$ there is room for some energy saving at a small price in throughput reduction, by choosing $t > m$. On the other hand, for $m = 3$, $t = m$ is already a very good choice, and further improvements are marginal.

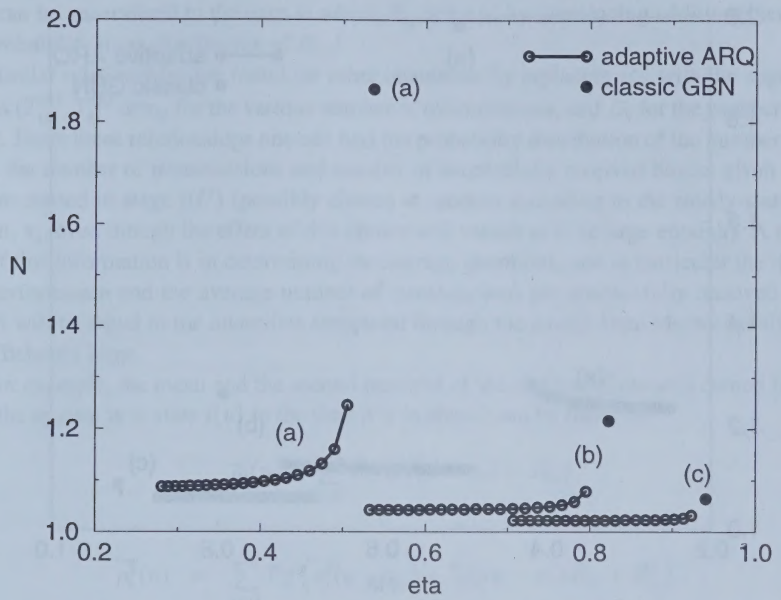


Figure 1: Average number of equivalent block transmissions, N, vs. the average throughput, η, for the classic GBN protocol and for the adaptive strategy, for $t = m$ to 20; power ratio $w = 0.2$; $m = 1$. (a) $\varepsilon = 0.271, r = 0.081$, (b) $\varepsilon = 0.095, r = 0.161$, (a) $\varepsilon = 0.031, r = 0.292$.

Of course, for better channel conditions, the classic ARQ will perform just as well, since most of the packets do not need to be retransmitted, and $1/\eta$ is already close to 1. The adaptive strategy, on the other hand, can be used in environments where the impairments caused by propagation and interference are dominant.

4 A more detailed approach

The technique described so far is adequate for finding the steady-state averages of the quantities of interest, but fails to track the detailed information needed to model the behavior of the batteries. In particular, the transition probabilities used above are independent of the status of the battery. This corresponds to a battery whose output energy is a constant as long as there is some charge in it, and drops to zero when the battery is discharged. More realistically, the decrease in the charge may not be a step function.

In order to capture these features, we expand the model in the following way. Consider the transition diagram of the Markov model described above, and add another dimension, u, which represents the units of energy that the battery has available. This results in a diagram composed of a number of stages, 0 through U, where stage 0 is composed of one state, which is an absorbing state (the battery is completely discharged). We assume for simplicity that $w = 1/K$, with K an integer, i.e., the energy required for a block transmission is an integer

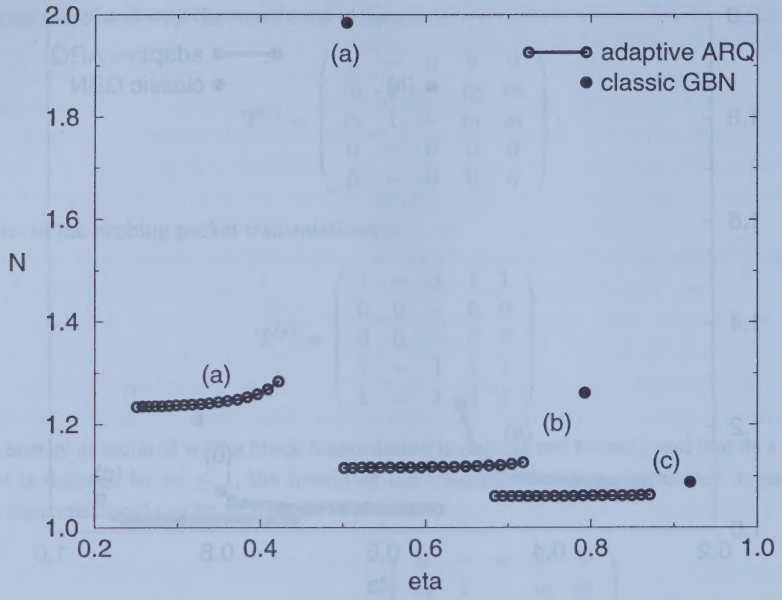


Figure 2: Average number of equivalent block transmissions, N, vs. the average throughput, η, for the classic GBN protocol and for the adaptive strategy, for $t = m$ to 20; power ratio $w = 0.2$; $m = 3$. (a) $\varepsilon = 0.271, r = 0.081$, (b) $\varepsilon = 0.095, r = 0.161$, (a) $\varepsilon = 0.031, r = 0.292$.

multiple of that for a probe transmission. In this case, the battery charge can be measured in terms of probe packet transmissions in our model. This can be generalized to the case in which w is any rational number, but a larger U, although still finite, is needed. Let

$$\boldsymbol{\tau} = \frac{1}{w}\mathbf{T} = K\mathbf{T}^{(b)} + \mathbf{T}^{(p)} \tag{18}$$

be the matrix of consumed energy per transition, measured in probe packet units.

The transition structure of the Markov model is still maintained, except for the fact that the appropriate stage for the destination must be identified. In particular, a transition from i to j (which occurs with probability P_{ij}) in the Markov chain gives rise to transitions from $i(u)$ to $j(u - \tau_{ij})$ in the flow graph (τ_{ij} energy units are consumed). Also, some transitions for the stages close to 0 have to be modified, since there are no stages for $u < 0$ (actually, one can think of all "negative" stages as collapsed into state 0). This allows us to write recursive equations for the quantities of interest, which can be defined as random variables associated with the protocol evolution.

For example, let $\xi_i(u, \rho)$ be the probability of having ρ rewards starting from state $i(u)$ to absorption. The general recursive relationship can be written as follows:

$$\xi_i(u, \rho) = \sum_{j \in \Omega} P_{ij}\xi_j(u - \tau_{ij}, \rho - R_{ij}) \tag{19}$$

where R_{ij} is the deterministic number of rewards earned in the ij-th transition. (This relationship can be generalized to the case in which R_{ij} is a r.v., by introducing additional terms and the probability mass distribution of R_{ij}.)

Similar relationships are found for other quantities, by replacing R_{ij} with the appropriate values ($T_{ij}^{(b)}$, $T_{ij}^{(p)}$ or τ_{ij} for the various number of transmissions, and D_i for the number of time slots). From these relationships one can find the probability distribution of the number of time slots, the number of transmissions and number of successfully received blocks given that the system started in stage $i(U)$ (possibly chosen at random according to the steady-state distribution, π_i, even though the effect of this choice will vanish as U is large enough). A possible use of this information is in determining the average quantities, and in particular the throughput performance and the average number of transmissions per successfully received packet, which will be equal to the quantities computed through the steady-state Markov analysis, for U sufficiently large.

For example, the mean and the second moment of the number of rewards earned from the time the system is in state $i(u)$ to the time it is in state 0 can be found as

$$\overline{\rho_i}(u) = \sum_{j \in \Omega} P_{ij} \left(\overline{\rho_j}(u - \tau_{ij}) + R_{ij} \right) \tag{20}$$

$$\overline{\rho_i^2}(u) = \sum_{j \in \Omega} P_{ij} \left(\overline{\rho_j^2}(u - \tau_{ij}) + 2\overline{\rho_j}(u - \tau_{ij}) R_{ij} + R_{ij}^2 \right). \tag{21}$$

From (20) and (21), it is possible to find mean and variance by averaging over i. Note that if U is large enough, and the stage-dependent parameters vary slowly with u, the steady-state distribution π_i can be used.

This approach is powerful, not only because it gives the complete statistical information, but also because it allows us to essentially have all quantities dependent on the energy state, u. For example, it is possible to incorporate performance degradation due to battery discharge, by choosing $P_{ij} = P_{ij}(u)$. Although this approach is a little tedious, it conserves enough modularity in the transition structure to allow efficient computations, once the transition branches have been specified. In particular, when the quantities involved are functions of u, recursive equations such as (19) still hold.

4.1 Battery recovery

There is evidence that some batteries have the ability to recover some of their charge when energy is not drained from them. For example, it is shown in [6] that the capacity of a cell depends on the manner in which it is discharged, and that parameters such as the frequency and the duty-cycle of a pulsed discharge regime have an impact on the battery life.

It is worth noting that the analysis presented here could incorporate this feature as well, with little additional complexity. For the sake of discussion, assume that a battery recovers α units per slot on the average. Since an energy unit can be used only when completely recovered, it is like having an energy upgrade every $1/\alpha$ slots. This can be included in the model by assuming that there is an energy upgrade with probability $D_i\alpha$ associated with transition ij, where D_i is the corresponding number of time slots. Note that this could be generalized by including transition-dependent upgrade probabilities, α_{ij} instead of $D_i\alpha$, to take into account the fact that different energy upgrades may be associated with silent slots, block transmissions

and probe transmissions. In this case, the steady-state analysis can be extended by introducing an appropriate "recovery" matrix to be subtracted from $\mathbf{T}$ in (16).

In the flow graph model, we can modify the recursive relationships to take into account the possibility that an energy unit is earned upon a transition. As an example, (19) can be rewritten as

$$\xi_i(u,\rho) = \sum_{j\in\Omega} P_{ij}\left[\alpha_{ij}\xi_j(u-\tau_{ij}+1,\rho-R_{ij}) + (1-\alpha_{ij})\xi_j(u-\tau_{ij},\rho-R_{ij})\right]. \qquad (22)$$

5 Conclusions

In this paper, we have addressed the problem of the energy efficiency of ARQ error control protocols over Markov channels. It has been shown that the classic ARQ strategies may lead to a considerable waste of energy, due to the high number of transmissions. An adaptive scheme, which slows down the transmission rate when the channel is impaired, saves energy, without a significant loss in throughput. Such scheme can be successfully employed for reliable data transmission on mobile radio links. Finally an analytical approach that can track more complex models for energy sources, including dynamic charge recovery, is proposed and discussed. It is shown that this more detailed approach is computationally more burdensome but is otherwise very powerful in tracking the performance of error control protocols under more elaborate energy models than a simple step function.

Future work based on a better understanding of battery characteristics in this environment remains to be undertaken. We anticipate the possibility of developing new protocols that optimize the energy utilization and the battery life. Finally, the issue of ARQ vs. FEC and of the maximization of the efficiency through a careful choice of hybrid error control protocols also remains to be addressed.

References

[1] M. Zorzi, R.R. Rao, "Energy constrained error control for wireless channels", in *Proc. IEEE GLOBECOM'96*, Nov. 1996.

[2] M. Zorzi, R.R. Rao, L.B. Milstein, "On the accuracy of a first-order Markov model for data block transmission on fading channels", in *Proc. IEEE ICUPC'95*, pp. 211-215, Nov. 1995.

[3] Y.J. Cho and C.K. Un, "Performance analysis of ARQ error controls under Markovian block error pattern", *IEEE Trans. Commun.*, vol. COM-42, pp. 2051-2061, Feb.-Apr. 1994.

[4] R.A. Howard, *Dynamic probabilistic systems*, John Wiley & Sons, 1971.

[5] S.H. Ross, *Stochastic processes*, John Wiley & Sons, 1983.

[6] E.J. Podlaha, H.Y. Cheh, "Modeling of cylindrical alkaline cells. VI: variable discharge conditions", *J. Electrochem. Soc.*, vol. 141, pp. 28-35, Jan. 1994.

Time Dependent Outages in Cellular Systems : Minimum Duration Outages

Narayan B. Mandayam, Pi-Chun Chen, and Jack M. Holtzman
WINLAB, Department of ECE, Rutgers University, Piscataway, NJ 08855-0909

Abstract

Traditionally, the notion of outage for cellular systems has been studied based on the signal level (or, more precisely, the signal-to-interference ratio (SIR)) falling below a required threshold. In real cellular environments, it is not the instantaneous drop of the signal strength below the threshold that determines outage. It is, in fact, the duration of time below a threshold that determines outage for cellular systems. Moreover, the static analysis of outage precludes the time correlation in the signals which is important in real systems owing to mobility, fading and power control. In this paper, we analyze "minimum duration" outages via the theory of level crossings, where outage is defined as an excursion of the SIR below a level for a certain minimum duration. These "minimum duration" outages have implications in redefining user capacity and handoff performance. We also extend the above notion to scenarios where there is diversity in reception, such as those pertaining to soft handoff.

1 Introduction

Analysis of the problems concerning cellular system performance encounters serious difficulties since temporal variations due to factors such as mobility, fading channels, and time correlations of signals in the system are not easy to model. However, the theory of level crossings [1, 2, 3] has been applied to study the dynamics of time-varying signals for a variety of diverse applications including, in a limited manner, mobile radio communications [4, 5, 6]. We introduce here using the theory of level crossings, a new twist on mobile radio systems performance, namely the dependence on time (or duration) of such performance measures. The specific performance measure we focus on in this paper is outage for such systems. We use exact and asymptotic results from the theory to analyze the effect of delays in system procedures such as the exchange of information and processing times on the overall system performance. The motivation is that a signal (or signal-to-interference ratio, SIR) going below a threshold for only a very short interval may have hardly any deleterious effect while a longer duration may result in loss of the call. A "real" outage does not occur instantaneously. Instead, if the SIR stays under a threshold sufficiently long, we are in the risk of losing all the information. In other words, the duration below the threshold, in fact, plays an important factor in determining an outage. Therefore, we incorporate the concept of "minimum duration," τ_m, into the definition of an outage event. An outage event is defined as the SIR going below a threshold, and staying longer than the "minimum duration" τ_m. We formulate this problem as a level crossing problem and analyze the average duration of an outage, frequency of an outage, and probability of an outage and the effect of τ_m on the above quantities. Further, we extend this formulation to that pertaining to the case of reception diversity. Thus, there is a minimum duration outage only when two signals are simultaneously below threshold for a minimum duration. The minimum duration outage is related to mobile speed, correlation distance, and time dependence of signals.

2 Preliminaries

The theoretical basis for this paper is the general theory of level crossings and initial work already using level crossings. We focus here on the following important level crossing results

that will be used to model and evaluate outage performance of mobile communication systems.

Consider $x(t)$, a zero mean Gaussian process, a sample path of which is shown in Figure 1. The level crossings of this process x(t) of an arbitrary level γ is of interest. Specifically, there are the following results from existing literature on level crossings (the technical conditions given in [3]).

Result 1 (Rate of Level Crossing) : An exact expression can be derived for the rate of level crossings of $x(t)$ of the level γ. Specifically, the rate of crossings is given as

$$\text{Rate of Crossings} = \frac{1}{\pi}\sqrt{\frac{\lambda}{\sigma^2}}\exp(-\frac{\gamma^2}{2\sigma^2}), \tag{1}$$

where $\sigma^2 = R(0)$, with R(t) being the covariance function of the process $x(t)$, and $\lambda = -\frac{d^2}{dt^2}R(t)\mid_{t=0}$ with the necessary condition that $\lambda < \infty$. The above requirement can be explicitly stated by requiring the covariance function $R(t)$ to be such that

$$R(t) = 1 - \frac{\lambda t^2}{2} + O(t^2) \text{ as } t \to 0 \tag{2}$$

The above covariance function can be translated to reflect the correlation in distance by using the mobile velocity and correlation distance profile of the mobile environment.

Result 2 (Asymptotic Poisson Result) : The level crossings of very high levels approach a Poisson process, i.e., as γ gets large, the points of upcrossings of $x(t)$ of the level γ are modeled well by a Poisson arrival process.

Result 3 (Parabolic Sample Paths) : The trajectory of a level crossing above a very high level is characterized by a parabola with a Rayleigh distributed parameter. This can be used in turn to determine the distribution of duration of such level crossings. The result is: Let $x(t)$ be a stationary zero mean, unit variance Gaussian process with covariance function $R(\tau)$ satisfying

$$-R''(\tau) = \lambda + O(|\log|\tau||^{-a}) \quad as\ \tau \to 0, \tag{3}$$

for some $a > 1$, and

$$R(\tau) \to 0 \text{ as } \tau \to \infty. \tag{4}$$

Then as $\gamma \to \infty$, excursions of $\{x(t)\}$ above γ behave asymptotically as

$$x(t) \sim \gamma + \xi t - \gamma\frac{\lambda t^2}{2}, \tag{5}$$

where ξ is a Rayleigh random variable with density function

$$P_\xi(z) = \frac{z}{\lambda}\exp(\frac{-z^2}{2\lambda}),\ z \geq 0. \tag{6}$$

Results 1 and 2 form the basis of [6], while Result 3 was used in [7]. In this paper, we use Results 1 and 3 in order to analyze "minimum duration" outages for cellular systems.

3 Problem Formulation

We formulate the problem of "minimum duration" outages as a level crossing problem. We are interested in finding the average duration of an outage, frequency of an outage, and probability of an outage. Also, it is necessary to determine the effect of different τ_m's on the

three quantities above. Usually, it is very difficult to analyze such a level crossing problem with a timer. Since the outage event can be considered as a rare event, it is possible to apply asymptotic results as were previously used successfully for handoff problems in [6] and [7]. Consider $S(t)$ to be the desired user's received signal power at the base station, and $I(t)$ to be the interference power received at the base station. The signal-to-interference ratio (SIR) is now a time-varying function given as

$$SIR(t) = \frac{S(t)}{I(t)}.$$

We define the following performance measures as follows.

Definition 1 *An outage event is defined as $SIR(t)$ going below a required threshold γ and staying below at least τ_m seconds. In mathematical form,*

$$SIR(t) < \gamma, \forall t \in [\tau_a, \tau_b], \text{ and } \tau_b - \tau_a \geq \tau_m \tag{7}$$

Definition 2 *Frequency of outage is defined as rate of outage events;*

$$f_{out} = \lim_{t\to\infty} \frac{\text{Total number of outages occurring in } [0,t]}{t} \tag{8}$$

Definition 3 *Probability of outage is the fraction of time that the system is in outage;*

$$P_{out} = \lim_{t\to\infty} \frac{\text{Total outage time in } [0,t]}{t} \tag{9}$$

The objective is to find the following performance measures,
1) Average duration of an outage event, $\bar{\tau}_{out}$,
2) Frequency of outage, f_{out},
3) Probability of outage, P_{out}
as functions of parameters such as minimum duration τ_m, mobile speed v, and correlation distance of shadow fading d_c. In order to apply the results on level crossings (given in Section 2), we need the second derivative of the correlation function at the origin. The commonly used correlation function [8] is of the form $\exp(-|\frac{d}{d_c}|)$. Clearly, this function does not have a second derivative at the origin, and such a structure for the correlation function leads to infinite number of crossings of any finite level [3, 9]. A related observation is also detailed in [10]. However, an inspection of data shows that it is only a rought fit for measurements and therefore, we will assume that the model for the correlation function of shadow fading in distance and time are given by

$$R_D(d) = \exp(-\frac{1}{2}\left(\frac{d}{d_c}\right)^2), \tag{10}$$

and

$$R_T(t) = \exp(-\frac{1}{2}\left(\frac{vt}{d_c}\right)^2), \tag{11}$$

respectively where where d_c = correlation distance, v = mobile velocity.
Remark :The above shows that more attention needs to be paid to measurements, and that level crossings should be included in those measurements.

In a dB scale, the SIR going below a level γ is

$$S(t) - I(t) < \gamma$$

We assume that $S(t)$ is lognormal (reflecting either shadow fading, or CDMA power control errors) and that the interference I(t) is fixed. (The results can be easily extended to the case of random interference.) In order to apply the level crossing results in Section 2, we convert the problem of going below a level into a problem of going above a level as

$$X'(t) > \gamma', \tag{12}$$

where $X'(t)$ is a zero mean, unit variance Gaussian process given as

$$X'(t) = \frac{E[S(t) - I] - [S(t) - I]}{\sigma}, \tag{13}$$

and

$$\gamma' = \frac{E[S(t) - I] - \gamma}{\sigma}, \tag{14}$$

where σ^2 is the correlation function of $S(t)$ evaluated at the origin, i.e., $R(0)$. Using the above formulation, we derive the following results the details of which are outlined in the Appendix. Firstly, the average duration of an outage event is given as

$$\overline{\tau}_{out} = \tau_m + \sqrt{\frac{\pi}{A}} Q(\sqrt{2A}\tau_m) \exp(A\tau_m^2), \tag{15}$$

where

$$A = \frac{(\mu_S - \mu_I - \gamma)^2 v^2}{8\sigma^2 d_c^2}$$

with μ_S and μ_I being the mean signal and interference levels in dB. Secondly, the frequency of outage can be derived as

$$f_{out} = \frac{v}{2\pi d_c} \exp\left(-A(\tau_m^2 + \frac{4d_c^2}{v^2})\right). \tag{16}$$

Using the relationship $P_{out} = f_{out}\overline{\tau}_{out}$, we see from (15) and (16) that the probability of outage is given as

$$P_{out} = \frac{\tau_m v}{2\pi d_c} e^{-A\left(\tau_m^2 + \frac{4d_c^2}{v^2}\right)} + \frac{v}{2d_c\sqrt{\pi A}} Q(\sqrt{2A}\tau_m) e^{-4\frac{Ad_c^2}{v^2}}. \tag{17}$$

4 Extension to the Case of Reception Diversity

In this section, we further apply the concept of minimum duration outage to a system with reception diversity. For example, in soft handoff where a mobile is in reception of two received signals from two different base stations, an outage occurs only when both SIR levels are below threshold. Minimum duration outage with reception diversity (or "Double outage") is defined as both received signals going below a required threshold and staying longer than the minimum duration simultaneously. For the system with reception diversity, we consider two received signals at the mobile having signal to interference ratios $SIR_1(t)$, and $SIR_2(t)$. We formulate the problem as a simultaneous level crossing problem. The performance measures are defined as follows:

Definition 4 *A double outage event is defined as two received signals $SIR_1(t)$, $SIR_2(t)$ experiencing minimum duration outage simultaneously, i.e.*

$$\max\{SIR_1(t), SIR_2(t)\} < \gamma,$$

$$\forall t \in [\tau_a, \tau_b], \quad \textit{and} \quad (\tau_b - \tau_a) \geq \tau_m$$

Definition 5 *Frequency of double outage is defined as the rate of outage events;*

$$F_d = \lim_{t \longrightarrow \infty} \frac{E\left[\begin{array}{c} \textit{Total number of double outages} \\ \textit{occurring in } [0,t] \end{array}\right]}{t}.$$

Definition 6 *Probability of double outage is the fraction of time that the system is in double outage:*

$$P_d = \lim_{t \longrightarrow \infty} \frac{E[\textit{Total time in double outage in } [0,t]]}{t}.$$

The analysis of double outage starts with results from single outage analysis. The probability of double outage, of interest, is a fairly rare event, say 1% of the time. Much more than 1% of single outage is needed to have 1% of double outage. In this case, a single outage event might not be such a rare event anymore, i.e., the asymptotic results might not work well in the range of interest. Fortunately, the exact mean time for a signal staying below a certain level γ is known. We modify the Rayleigh model by assuming the mean of the Rayleigh random variable equals the exact mean time for the signal staying below a level. A similar approach is also given in [4]. Simulation results are used to verify this and other assumptions. Now the modified probability density function for duration τ has the following form:

$$f_\tau(\tau) = A\tau e^{\frac{-A\tau^2}{2}}, \tag{18}$$

and mean of τ; $\sqrt{\frac{\pi}{2A}}$ satisfies

$$\sqrt{\frac{\pi}{2A}} = \frac{2\pi d_c}{v} e^{\frac{\gamma^2}{2}} Q(\gamma) \tag{19}$$

where v is the mobile velocity, and d_c is the correlation distance of the signal. Due to page constraints, we refer the reader to [11] for a detailed analysis of the case of double outages.

5 Numerical Results

We first begin by validating the asymptotic analysis for outage considerations in cellular systems. To show the credibility of using asymptotic results, we consider the level crossings of the signal about a level γ with the minimum duration $\tau_m = 0$. The exact expression of the mean time above a level could be obtained by the ratio of the probability that the signal is above the level to the rate of level up-crossing. Also, we could calculate the approximation for mean time above a level using asymptotic parabolas. We compare the results from these two approaches to show that the outages may be considered as rare events for applicability of asymptotic results. A zero mean, unit variance normal random process is adopted in the comparison. The exact and asymptotic expressions for mean time above a level γ are in the following forms respectively:

$$\overline{\tau}_{out}^{exact} = \frac{2\pi}{\sqrt{\lambda}} e^{\frac{\gamma^2}{2}} Q(\gamma), \tag{20}$$

and

$$\overline{\tau}_{out}^{asymp.} = \sqrt{\frac{2\pi}{\lambda}} \frac{1}{\gamma}. \tag{21}$$

When the level is above about two times of the standard deviation, there is good agreement between exact analysis and asymptotic approximation as shown in Figure 2. These parameters are consistent with the low probability of outages that are being sought in a practical system.

The asymptotic approximation shown in Figure 3 shows good agreement to simulation within the confidence interval. Figure 4 and Figure 5 demonstrate the effects on frequency of outage and probability of outage for different τ_m's. In Figure 4, the frequency of outage decreases with the increasing of the minimum duration, and the family of curves have intersections with each other. For fixed mobile velocity, the higher values correspond to the less correlated signals, which is also responsible for the fast fading phenomenon. Examining the crosspoint of any two curves, it shows that with small τ_m, fast fading causes higher frequency of outage than slow fading. The reason is fast (shadow) fading has a higher frequency of level crossings. But with the increasing of τ_m, slow fading is more likely to develop into an outage. The situation is inverted as a consequence. In Figure 5, it is not surprising to find that the probability of outage decreases as τ_m increases. Moreover, for fixed τ_m and mobile velocity, strongly correlated signals have a higher probability of outage than that of relatively less correlated signals. Another observation is that for a fixed correlation distance, the mobile with lower velocity has a higher probability of outage.

For the two received signals case, the received SIR strengths are assumed to be the same, i.e. $SIR_1 = SIR_2$, and also possess the same autocorrelation distances. Figure 6 shows frequency of double outage as a function of τ_m. The simulation and the analysis are very well matched. Further, results in [11] demonstrate the effects on the probability of double outage due to different τ_m's with respect to different signal levels. It is shown that there is fairly good agreement between the analysis and the simulations. Note that probability of double outage is the product of frequency of double outage and average duration of double outage. To simplify the analysis of average duration of double outage, we assume that at most one double outage can occur during the period that one of the signals is in single outage.

6 Discussion and Conclusion

In this paper, we introduced the notion of "minimum duration" outages for a cellular system. The motivation behind this was that the time dependence of signals is all important in a mobile environment, and as a result a momentary drop of signal strength below a required threshold may not have any impact on the outage performance of the system. It is the duration of such a drop that is relevant in practical cellular systems. Asymptotic results from the theory of level crossings were used in deriving analytical results for the outage performance of such systems. These were further validated via exact analysis (where applicable) and simulations. These "minimum duration" outages were further related to fading conditions through mobile speeds and correlation distance profiles. It was shown that for practical values of outage probabilities, the results were accurate and these results could be used to make more realistic evaluations of system performance than existing static methods.

Extending the results presented here will lead to new definitions of capacity for such cellular systems. The introduction of minimum durations in outages will now redefine the user capacity of such systems in terms of the number of users that can be supported per cell. Based on realistic values of such minimum durations, we propose to evaluate comparisons of capacities of systems using different access technologies like CDMA and TDMA. Existing comparisons of the relative capacities are sensitive to numbers chosen for arguing the relative merits. We believe that the relative minimum durations that each type of system can tolerate may significantly redefine notions of user capacity and comparative analysis of the different technologies.

The introduction of "minimum duration outage" for the case of reception diversity provides a more realistic performance measure for a cellular system. An application of double outage of minimum duration is in describing the soft handoff outage where the

mobile can choose the stronger signal among these two. With this paper and [11], one can see how the new definitions affect comparisons of outages and handoff (soft and hard) algorithms. This is future work, along with further development of the analysis.

Appendix

We will derive the expressions for $\overline{\tau}_{out}$, f_{out}, and P_{out} that were presented in Section 3. Recall from (12) that the SIR (in dB scale) going below a level was formulated as

$$X^{'}(t) > \gamma^{'},$$

where $X^{'}(t)$ is a zero mean, unit variance Gaussian process given as

$$X^{'}(t) = \frac{E[S(t) - I] - [S(t) - I]}{\sigma}, \tag{22}$$

and

$$\gamma^{'} = \frac{E[S(t) - I] - \gamma}{\sigma}, \tag{23}$$

where σ^2 is the correlation function of $S(t)$ evaluated at the origin, i.e., $R(0)$. Using Result 3, the interval τ_a, that the process $X^{'}(t)$ is above the threshold $\gamma^{'}$, is a random variable given by

$$\tau_a = \frac{2}{\gamma^{'}\lambda}\zeta, \tag{24}$$

where ζ denotes the Rayleigh random variable whose pdf is given in (6). The pdf of τ_a is now given by

$$f_{\tau_a}(\tau_a) = \frac{\lambda\gamma^{'2}\tau_a}{4}\exp\left(\frac{-\lambda\gamma^{'2}\tau_a^2}{8}\right).$$

From the above, the pdf of the outage duration τ_{out} can be derived as

$$\begin{aligned} f_{\tau_{out}}(\tau_{out}) &= \begin{cases} \frac{f_{\tau_a}(\tau_{out})}{P(\tau_a > \tau_m)} & \tau_{out} > \tau_m \\ 0 & \text{otherwise} \end{cases} \\ &= \begin{cases} f_{\tau_a}(\tau_{out})\exp\left(\frac{\lambda\gamma^{'2}\tau_m^2}{8}\right) & \tau_{out} > \tau_m \\ 0 & \text{otherwise} \end{cases} \end{aligned}$$

where

$$P(\tau_a > \tau_m) = \int_{\tau_m}^{\infty} f_{\tau_a} d\tau_a = \exp(-\frac{\lambda\gamma^{'2}\tau_m^2}{8})$$

The average duration of outage is now given as

$$\begin{aligned} \overline{\tau}_{out} &= \int_{-\infty}^{\infty} \tau_{out} f_{\tau_{out}} d\tau_{out} \\ &= \int_{\tau_m}^{\infty} \frac{\tau_a f_{\tau_a}(\tau_a)}{P(\tau_a > \tau_m)} d\tau_a \\ &= \exp\left(\frac{-\lambda\gamma^{'2}\tau_m^2}{8}\right) \int_{\tau_m}^{\infty} \tau_a f_{\tau_a}(\tau_a) d\tau_a \\ &= \exp(A\tau_m^2)\left[\tau_m \exp(-A\tau_m^2) + \sqrt{\frac{\pi}{A}} Q(\sqrt{2A}\tau_m)\right], \end{aligned}$$

where

$$A = \frac{\lambda \gamma'^2}{8}.$$

The frequency of outage is given as

$$f_{out} = f_{up-crossing} \times \text{Prob}\{\text{the crossing is an outage}\},$$

where $f_{up-crossing}$ is given by Result 1 as

$$f_{up-crossing} = \frac{\sqrt{\lambda}}{2\pi} \exp\left(-\frac{\gamma'^2}{2}\right),$$

and

$$\text{Prob}\{\text{the crossing is an outage}\} = P(\tau_a > \tau_m).$$

Therefore the frequency of the outage is

$$f_{out} = \frac{\sqrt{\lambda}}{2\pi} \exp\left(-A\left[\tau_m^2 + \frac{4}{\lambda}\right]\right).$$

To represent $\overline{\tau}_{out}$, f_{out}, P_{out} as functions of the mobile velocity v and of the correlation distance of shadow fading d_c, we find that the relationship between λ, v, and d_c is given as

$$\lambda = \left(\frac{v}{d_c}\right)^2.$$

References

[1] I.F. Blake and W.C. Lindsey, "Level-Crossing Problems for Random Processes," *IEEE Transactions on Information Theory*, Vol. IT-19, No.3, pp 295-315, May 1973.

[2] J. Abrahams, "A Survey of Recent Progress on Level-Crossing Problems for Random Processes," in *Communications and Networks - A Survey of Recent Advances*, Ian F. Blake and H. Vincent Poor, eds., Springer-Verlag, New York, 1986.

[3] M.R. Leadbetter, G. Lindgren, and H. Rootzen, "Extremes and Related Properties of Random Sequences and Processes," New York: Springer-Verlag, 1983.

[4] S.O. Rice, "Distribution of the Duration of Fades in Radio Transmissions," *Bell System Technology Journal*, Vol.37, pp. 581-635, May 1958.

[5] M. D. Austin and G.L. Stüber, "Velocity Adaptive Handoff Algorithms for Microcellular Systems," *IEEE Transactions on Vehicular Technology*, Vol.43, No.3, pp. 549-561, August 1994.

[6] R. Vijayan and J. M. Holtzman, "A Model for Analyzing Handoff Algorithms," *IEEE Transactions on Vehicular Technology*, Vol.42, No. 3, pp. 351-356, August 1993.

[7] R. Vijayan and J. M. Holtzman, "Foundations for Level Crossing Analysis of Handoff Algorithms," *Proceedings of IEEE ICC'93*, Geneva, pp. 935-939, May 1993.

[8] M. Gudmundson, "Correlation Model for Shadow Fading in Mobile Radio Systems," *Electronic Letters*, Vol. 27, No. 23, pp. 2145-2146, November 1991.

[9] M.R. Leadbetter, "Point Processes Generated by Level Crossings," in *Stochastic Point Processes*, P.A.W. Lewis, Ed., New York, NY: Wiley, pp. 436-467, 1972.

[10] A.J. Rainal, "Bandwidth from a Large Excursion of Gaussian Noise," *IEEE Transactions on Instrument Measures*, Vol. 40, pp. 688-693, August 1991.

[11] P. C. Chen, N. Mandayam, J. Holtzman, "Minimum Duration Outage for Cellular Systems with Reception Diversity," *Proc. of IEEE ICUPC'96*, September, 1996

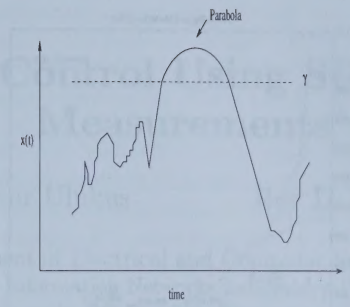


Figure 1: Sample Path of $x(t)$

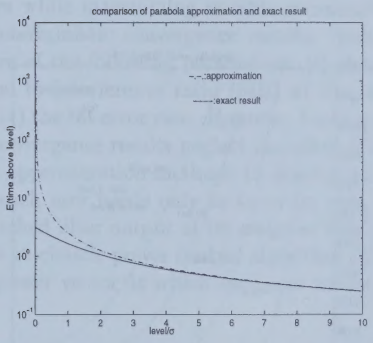


Figure 2: Comparison of Asymptotic and Exact Results

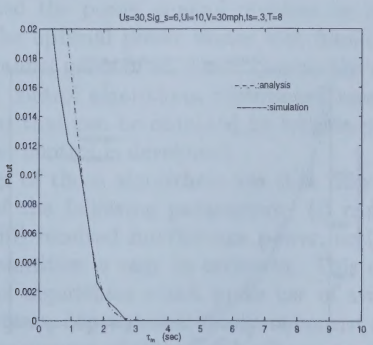


Figure 3: Comparison with Simulation

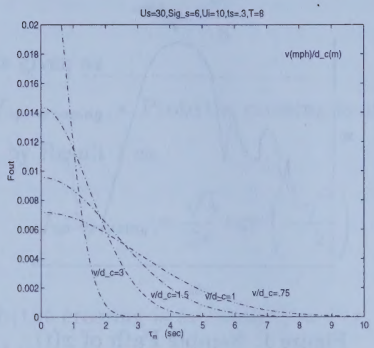


Figure 4: Frequency of Outage

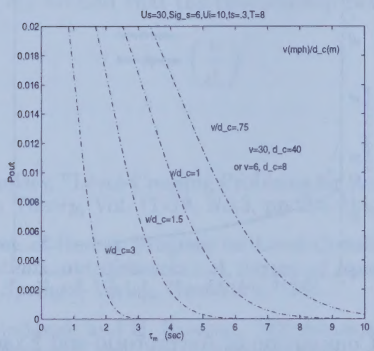


Figure 5: Probability of Outage

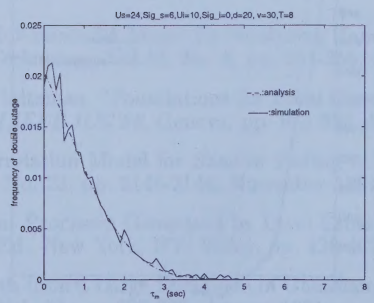


Figure 6: Analytical and simulation results for double outage

Power Control Using Stochastic Measurements*

Sennur Ulukus Roy D. Yates

Department of Electrical and Computer Engineering
Wireless Information Networks Laboratory (WINLAB)
Rutgers University, PO Box 909, Piscataway NJ 08855-0909
e-mails: *ulukus@winlab.rutgers.edu, ryates@winlab.rutgers.edu*

Abstract

For wireless communication systems, iterative power control algorithms have been proposed to minimize transmitter powers while maintaining reliable communication between mobiles and base stations. To derive deterministic convergence results, these algorithms require perfect measurements of one or more of the following parameters: (i) channel gains of some or all the users, (ii) the mobile's signal to interference ratio (SIR) at the receiver, (iii) the interference experienced by the mobile, (iv) the bit error rate. However, these quantities are often difficult to measure and deterministic convergence results neglect the effect of stochastic measurements. In this work, we use stochastic approximation methods to develop an iterative, distributed power control algorithm in which each user needs only to know its own channel gain to its assigned base station and its own matched filter output at its assigned base station. If the power control problem is feasible, then this stochastic power control algorithm (SPCA) converges in the mean square sense to an optimal power vector at which each user meets an SIR objective.

1 Introduction

Power control has been an important part of resource management in cellular wireless communication systems. The aim of power control is to assign each user a transmitter power level such that all users in the system satisfy their quality of service requirements. Earlier work [1, 2] identified the power control problem as an eigenvalue problem for non-negative matrices. The optimal power vector was found by inversion of a matrix which was composed of channel gains of all users. Due to the computational complexity of these centralized power control algorithms, distributed versions have been developed which need only path gains that can be obtained by local measurements [3–5]. In [6], a framework for uplink power control is developed.

The main shortcomings of those algorithms are that they require knowledge or estimates of one or more of the following parameters: (i) channel gains, (ii) signal to interference ratio (SIR), (iii) received interference power, or (iv) bit error rate. Unfortunately, none of those quantities is easy to estimate. This observation highlights the need for new power control algorithms which make use of available or easily estimable parameters. In CDMA systems, conventional receivers consist of matched filters that are matched to the signature sequences of the users in the system. We propose an iterative and distributed stochastic power control algorithm (SPCA) which is based on the observation of the outputs of the matched filters that are available at the receivers. Each

*Supported by NSF Grant NCR 95-06505

user needs to know only its own matched filter output at its assigned base station and its own channel gain to its assigned base station to run the proposed algorithm. The SPCA proposed in this paper is based on the idea of stochastic approximations.

The method of stochastic approximation was first introduced by Robbins and Monro [7], who solved a deterministic problem with unknown parameters by observing the random outputs of a controllable experiment [8, 9]. In the application of stochastic approximation methods to the power control problem, the random experiments are the transmissions of information bits from users to the base stations. Those experiments are stochastic due to the randomness of the transmitted bits and to the existence of additive white Gaussian noise and multiuser interference. The experiments are controlled by the iterative update of the transmitter powers of the active users.

In this work we consider the uplink of a wireless cellular system with synchronous CDMA scheme and BPSK modulation and a fixed assignment of N active users to M base stations. We assume that the base stations have conventional receivers.

2 Deterministic Power Control

The aim of a power control algorithm is to minimize the users' transmitted powers while maintaining a certain quality of service (QOS) which is defined in terms of the Signal to Interference Ratio (SIR) of the individual users. We can state the power control problem as

$$\begin{array}{ll} \min & \sum_{i=1}^{N} p_i \\ \text{s.t.} & \frac{p_i h_{ii}}{\sum_{j \neq i} p_j h_{ij} \Gamma_{ij}^2 + \sigma^2} \geq \gamma^* \qquad \text{for } i = 1, \ldots, N \end{array} \tag{1}$$

where we used p_i to the power of user i, h_{ij} to denote the channel gain of user j to the assigned base station of user i, Γ_{ij} to denote the cross-correlation between the signature sequences of users i and j and γ^* to denote the common SIR *target* for all the users. Define $\boldsymbol{A} = [A_{ij}]$ and $\boldsymbol{H} = [H_{ij}]$ as

$$A_{ij} = \begin{cases} 0 & \text{if } i = j \\ \Gamma_{ij}^2 h_{ij} & \text{if } i \neq j \end{cases} \qquad \text{and} \qquad H_{ij} = \begin{cases} h_{ii} & \text{if } i = j \\ 0 & \text{if } i \neq j \end{cases} \tag{2}$$

Using the transmitted power vector $\boldsymbol{p}$ with ith element p_i, the QOS requirements in Equation (1) can be written as the vector inequality

$$\boldsymbol{p} \geq \gamma^* \boldsymbol{H}^{-1}(\boldsymbol{A}\boldsymbol{p} + \sigma^2 \boldsymbol{u}) \tag{3}$$

where $\mathbf{u} = [1 \quad 1 \quad \cdots \quad 1]^T$. We say that the SIR target γ^* is feasible if there is a non-negative finite vector $\boldsymbol{p}$ that satisfies (3).

It is not difficult to show that if the SIR target (γ^*) is feasible then the power vector which satisfies all N inequalities as equality in Equation (1) minimizes the sum of the transmitted powers (see [6]). Therefore, if γ^* is feasible, a power control algorithm finds (in the iterative case converges to) the optimal power vector, $\bar{\boldsymbol{p}}$, which is given as

$$\bar{\boldsymbol{p}} = \sigma^2 \gamma^* \boldsymbol{B}^{-1} \boldsymbol{H}^{-1} \boldsymbol{u} \qquad \text{where} \qquad \boldsymbol{B} = \boldsymbol{I} - \gamma^* \boldsymbol{H}^{-1} \boldsymbol{A} \tag{4}$$

The existence of a non-negative solution $\bar{\boldsymbol{p}}$ is specified by the following two theorems on eigenvalues of non-negative matrices by Perron and Frobenius [10].

Theorem 1 *A square non-negative matrix $\boldsymbol{S}$ has an eigenvalue $\rho = \rho(\boldsymbol{S})$ called the Perron-Frobenius eigenvalue of $\boldsymbol{S}$, such that (a) ρ is real and non-negative (b) with ρ can be associated non-negative left/right eigenvectors (c) $\rho \geq |\lambda|$ for any eigenvalue λ of $\boldsymbol{S}$.*

Theorem 2 *For an irreducible non-negative matrix* $\boldsymbol{S}$, $\boldsymbol{x} = \boldsymbol{S}\boldsymbol{x} + \boldsymbol{c}$ *has a non-negative solution* $\boldsymbol{x}$ *for any non-negative, nonzero* $\boldsymbol{c}$ *iff* $\rho(\boldsymbol{S}) < 1$.

From Theorem 2, we see that $\bar{\boldsymbol{p}}$ is non-negative iff $\rho(\gamma^* \boldsymbol{H}^{-1}\boldsymbol{A}) < 1$. Equivalently, we can make the following claim.

Lemma 1 *The SIR target* γ^* *is feasible iff* $\gamma^* < 1/\rho(\boldsymbol{H}^{-1}\boldsymbol{A})$.

If the correlation coefficients and channel gains are known and the SIR target is feasible, then the optimal power vector can be found by using Equation (4). However, using Equation (4) to solve the power control problem requires the exact knowledge of all channel gains and correlation coefficients of all the users in the system and the inversion of an $N \times N$ matrix.

Iterative and distributed deterministic power control algorithms can be defined in general as [6],

$$\boldsymbol{p}(n+1) = \boldsymbol{I}(\boldsymbol{p}(n)) \tag{5}$$

If the SIR target γ^* is feasible, then the algorithm (5) will converge to the optimal solution $\bar{\boldsymbol{p}} = \boldsymbol{I}(\bar{\boldsymbol{p}})$ if $\boldsymbol{I}(\boldsymbol{p})$ is a *standard* interference function; see [6]. We can express the interference vector using the notation of the present paper as,

$$\boldsymbol{I}(\boldsymbol{p}) = \gamma^* \boldsymbol{H}^{-1}(\boldsymbol{A}\boldsymbol{p} + \sigma^2 \boldsymbol{u}) \tag{6}$$

When $\boldsymbol{I}(\boldsymbol{p})$ is of the form of (6), the iteration (5) is the power control algorithm of Foschini and Miljanic [5, Eqn. (18)]. In the next section, we will exploit the close relation between the squared matched filter outputs and the interference function $\boldsymbol{I}(\boldsymbol{p})$ to introduce stochastic power control.

3 Stochastic Power Control

The matched filter output of the ith user at its assigned base station is,

$$y_i = \sum_{j=1}^{N} \sqrt{p_j}\sqrt{h_{ij}}\Gamma_{ij} b_j + n_i \tag{7}$$

where b_j is the information bit transmitted by user j ($+1$ or -1 with equal probability) and n_i is the additive white Gaussian noise (AWGN) random variable with zero mean and σ^2 variance.

Let us define an N dimensional random vector $\boldsymbol{v} = [\ y_1^2 \quad y_2^2 \cdots \quad y_N^2\ ]^T$ whose ith element is the squared value of the matched filter output y_i of user i at its assigned base station at the end of an arbitrary bit interval. Squaring both sides of Equation (7) yields

$$v_i = y_i^2 = \sum_{j=1}^{N} \Gamma_{ij}^2 h_{ij} p_j + w_i \tag{8}$$

where the contribution of the noise to the squared correlator output is

$$w_i = \sum_{j=1}^{N}\sum_{k\neq j} \sqrt{p_j}\sqrt{p_k}\Gamma_{ij}\Gamma_{ik}\sqrt{h_{ij}}\sqrt{h_{ik}} b_j b_k + 2n_i \sum_{j=1}^{N} \Gamma_{ij}\sqrt{p_j}\sqrt{h_{ij}} b_j + n_i^2 \tag{9}$$

Equivalently, in terms of the vectors $\boldsymbol{v}$, $\boldsymbol{p}$ and a noise vector $\boldsymbol{w}$ with ith component is w_i,

$$\boldsymbol{v} = (\boldsymbol{A} + \boldsymbol{H})\boldsymbol{p} + \boldsymbol{w} \tag{10}$$

Note that $E[w_i] = \sigma^2$, since the transmitted bits are independent and have equal probability of being $+1$ or -1, and the zero mean noise n_i is independent of the transmitted bits and has variance σ^2. Hence we obtain,

$$E[\boldsymbol{w}] = \sigma^2 \boldsymbol{u} \qquad \text{and} \qquad E[\boldsymbol{v}] = (\boldsymbol{A} + \boldsymbol{H})\boldsymbol{p} + \sigma^2 \boldsymbol{u} \tag{11}$$

From (11) we see that the expected value $E[\boldsymbol{v}]$ of the squares of the matched filter outputs is equal to the total received power. Thus each v_i can be viewed as an unbiased estimate of the total received power through the matched filter for user i at its assigned base station. Applying (11) to (6) yields

$$\boldsymbol{I}(\boldsymbol{p}) = -\gamma^* \boldsymbol{p} + \gamma^* \boldsymbol{H}^{-1} E[\boldsymbol{v}] \tag{12}$$

Equation (12) basically subtracts the signal component from the total received power to yield the total interference.

In [6], the following deterministic power control algorithm based on interference averaging was given.

$$\boldsymbol{p}(n+1) = (1-\epsilon)\boldsymbol{p}(n) + \epsilon \boldsymbol{I}(\boldsymbol{p}(n)) \tag{13}$$

In the algorithm (13), the required power $\boldsymbol{I}(\boldsymbol{p}(n))$ at iteration n is averaged with the current power $\boldsymbol{p}(n)$ to yield the new power vector $\boldsymbol{p}(n+1)$. Note that if $\epsilon = 1$ then (13) reduces to (5). Using the notation of the present paper, the interference averaging algorithm (13) is

$$\boldsymbol{p}(n+1) = (1-\epsilon)\boldsymbol{p}(n) + \epsilon \left(-\gamma^* \boldsymbol{p}(n) + \gamma^* \boldsymbol{H}^{-1} E[\boldsymbol{v}(n)]\right) \tag{14}$$

Replacing the total received power $E[\boldsymbol{v}]$ by the estimate $\boldsymbol{v}$ in the deterministic iteration (14) yields the following SPCA,

$$\boldsymbol{p}(n+1) = (1-\epsilon)\boldsymbol{p}(n) + \epsilon(-\gamma^* \boldsymbol{p}(n) + \gamma^* \boldsymbol{H}^{-1} \boldsymbol{v}(n)) \tag{15}$$

We note that Equation (15) is a special case of the following more general algorithm,

$$\boldsymbol{p}(n+1) = [1 - a_n(1+\gamma^*)]\boldsymbol{p}(n) + a_n \gamma^* \boldsymbol{H}^{-1} \boldsymbol{v}(n) \tag{16}$$

In the special case shown in Equation (15), the sequence a_n was taken to be a constant number $a_n = \epsilon$, independent of the iteration index n. From Equation (10), we have at time n, $\boldsymbol{v}(n) = (\boldsymbol{A} + \boldsymbol{H})\boldsymbol{p}(n) + \boldsymbol{w}(n)$. Applying this result to Equation (16), we obtain

$$\boldsymbol{p}(n+1) = \boldsymbol{p}(n) - a_n \left[\boldsymbol{B}\boldsymbol{p}(n) - \gamma^* \boldsymbol{H}^{-1} \boldsymbol{w}(n)\right] \tag{17}$$

It will be convenient to define

$$\boldsymbol{\eta}(n) = \gamma^* \boldsymbol{H}^{-1} \left(\boldsymbol{w}(n) - \sigma^2 \boldsymbol{u}\right) \tag{18}$$

Since (11) implies $E[\boldsymbol{\eta}(n)] = 0$, we see that $\boldsymbol{\eta}(n)$ represents a normalized form of the random component of the noise contribution $\boldsymbol{w}(n)$. Note that $\boldsymbol{\eta}(n)$ depends on the power vector components, transmitted bits and AWGN samples. Applying (18) to (17) and observing from (4) that $\boldsymbol{B}\bar{\boldsymbol{p}} = \sigma^2 \gamma^* \boldsymbol{H}^{-1} \boldsymbol{u}$, we obtain

$$\boldsymbol{p}(n+1) = \boldsymbol{p}(n) - a_n[\boldsymbol{B}(\boldsymbol{p}(n) - \bar{\boldsymbol{p}}) - \boldsymbol{\eta}(n)] \tag{19}$$

The convergence of (19) is stochastic since $\boldsymbol{\eta}(n)$ depends on the Gaussian noise and transmitted bits. The convergence of (19) will be specified in terms of the mean squared

error at iteration n,

$$\mathrm{MSE}(n) = E\left[\|\boldsymbol{p}(n) - \bar{\boldsymbol{p}}\|^2\right] \tag{20}$$

We will prove that if the deterministic power control problem is feasible, then under certain conditions on a_n, the sequence $\boldsymbol{p}(n)$ converges to the optimal power vector $\bar{\boldsymbol{p}}$ in the mean square sense that $\lim_{n\to\infty} \mathrm{MSE}(n) = 0$.

4 Stochastic Convergence Results

In this section, we will derive mean squared convergence results starting with the most general form of the stochastic power control iteration (19).

Although at the end we will prove convergence in terms of the Euclidean norm, the lack of symmetry in the system (in particular, $\boldsymbol{B}$ is not symmetric), dictates that we start our proof with a specific type of norm. After deriving the necessary results using this norm, we will go back to the Euclidean norm by using the result of Lemma 2 given below.

The definition of a *stable* matrix given below can be found in [11, page 403].

Definition 1 *A real matrix $\boldsymbol{S}$ is stable iff all of its eigenvalues have negative real parts.*

We also state the classical result of Lyapunov; see [11, page 405].

Theorem 3 *A matrix $-\boldsymbol{S}$ is stable iff for any positive definite symmetric matrix $\boldsymbol{C}$ there exists a positive definite symmetric matrix $\boldsymbol{G}$ such that,*

$$\boldsymbol{S}^\top \boldsymbol{G} + \boldsymbol{G}\boldsymbol{S} = \boldsymbol{C} \tag{21}$$

The following theorem states a simple result from linear algebra.

Theorem 4 *If $\boldsymbol{G}$ is a symmetric and positive definite matrix then $\|\boldsymbol{x}\|_G = (\boldsymbol{x}^\top \boldsymbol{G}\boldsymbol{x})^{1/2}$ is a norm. Denoting the smallest and the largest eigenvalues of $\boldsymbol{G}$ as $\lambda^G_{\min}$ and $\lambda^G_{\max}$,*

$$\lambda^G_{\min}\|\boldsymbol{x}\|^2 \le \|\boldsymbol{x}\|^2_G \le \lambda^G_{\max}\|\boldsymbol{x}\|^2 \qquad \text{and} \qquad \frac{1}{\lambda^G_{\max}}\|\boldsymbol{x}\|^2_G \le \|\boldsymbol{x}\|^2 \le \frac{1}{\lambda^G_{\min}}\|\boldsymbol{x}\|^2_G \tag{22}$$

First, we observe that a simple consequence of (22) is the following lemma.

Lemma 2 *If $\lim_{n\to\infty} E\left[\|\boldsymbol{p}(n) - \bar{\boldsymbol{p}}\|^2_G\right] = 0$, then $\lim_{n\to\infty} E\left[\|\boldsymbol{p}(n) - \bar{\boldsymbol{p}}\|^2\right] = 0$.*

We will assume that the deterministic power control problem is feasible in the sense that the optimal power vector $\bar{\boldsymbol{p}}$ of Equation (4) exists. This implies that all eigenvalues of $\gamma^* \boldsymbol{H}^{-1}\boldsymbol{A}$ have magnitude less than 1. By (4), all eigenvalues of $-\boldsymbol{B}$ must have negative real parts, yielding the following claim.

Lemma 3 *The matrix $-\boldsymbol{B}$ is stable iff the deterministic power control problem (1) is feasible.*

To simplify our convergence proofs, we define $\boldsymbol{x}(n) = \boldsymbol{p}(n) - \bar{\boldsymbol{p}}$ and we will verify that $\boldsymbol{x}(n)$ converges to the zero vector. We start our convergence proof with Equation (19) and (i) subtract $\bar{\boldsymbol{p}}$ from both sides; (ii) take and square the G-norms of both sides; (iii) take the conditional expectation of both sides, conditioned on $\boldsymbol{x}(n) = \boldsymbol{x}$; and (iv) observe that $E[\boldsymbol{\eta}(n) \mid \boldsymbol{x}(n) = \boldsymbol{x}] = \boldsymbol{0}$, to obtain

$$\begin{aligned} E\left[\|\boldsymbol{x}(n+1)\|^2_G \mid \boldsymbol{x}(n) = \boldsymbol{x}\right] &= \|\boldsymbol{x}\|^2_G - a_n \boldsymbol{x}^\top \left[\boldsymbol{G}\boldsymbol{B} + \boldsymbol{B}^\top \boldsymbol{G}\right]\boldsymbol{x} + a_n^2 \boldsymbol{x}^\top \boldsymbol{B}^\top \boldsymbol{G}\boldsymbol{B}\boldsymbol{x} \\ &\quad + a_n^2 E\left[\|\boldsymbol{\eta}(n)\|^2_G \mid \boldsymbol{x}(n) = \boldsymbol{x}\right] \end{aligned} \tag{23}$$

Our proof of convergence will proceed by bounding the individual terms on the right hand side of (23). As proven in Lemma 3, the feasibility of the deterministic power control problem guarantees the stability of $-\boldsymbol{B}$. By Theorem 3, stability of matrix $-\boldsymbol{B}$, in turn, guarantees the existence of a symmetric, positive definite matrix $\boldsymbol{G}$ as a solution of $\boldsymbol{GB} + \boldsymbol{B}^\top \boldsymbol{G} = \boldsymbol{C}$ for any symmetric and positive definite $\boldsymbol{C}$. Therefore, selection of the G-norm allows us to develop bounds for the second term in (23).

Lemma 4 *There exist positive constants k_0 and k_0', $0 < k_0 \leq k_0' < \infty$ and k_1 and k_1', $0 < k_1 \leq k_1' < \infty$ such that*

$$k_0 \|\boldsymbol{x}\|_G^2 \leq \boldsymbol{x}^\top \left[\boldsymbol{GB} + \boldsymbol{B}^\top \boldsymbol{G}\right] \boldsymbol{x} \leq k_0' \|\boldsymbol{x}\|_G^2 \tag{24}$$

$$k_1 \|\boldsymbol{x}\|_G^2 \leq \boldsymbol{x}^\top \boldsymbol{B}^\top \boldsymbol{GB} \boldsymbol{x} \leq k_1' \|\boldsymbol{x}\|_G^2 \tag{25}$$

Lemma 4 is a trivial consequences of Theorem 4 and the fact that both $\boldsymbol{GB} + \boldsymbol{B}^\top \boldsymbol{G}$ and $\boldsymbol{B}^\top \boldsymbol{GB}$ are symmetric, positive definite matrices. The next claim, Lemma 5, on the other hand, is more difficult to prove because of the indirect way in which the power vector $\boldsymbol{p}(n)$ interacts with the noise vector $\boldsymbol{\eta}(n)$. See [12] for proofs of these lemmas.

Lemma 5 *There exist positive constants d_0, c_0 and c_1, $0 < d_0 \leq c_0 < \infty$, such that*

$$d_0 \leq E\left[\|\boldsymbol{\eta}(n)\|_G^2 \mid \boldsymbol{x}(n) = \boldsymbol{x}\right] \leq c_0 + c_1 \|\boldsymbol{x}\|_G^2 \tag{26}$$

Using the results of Lemmas 4 and 5, we obtain

$$E\left[\|\boldsymbol{x}(n+1)\|_G^2 \mid \boldsymbol{x}(n) = \boldsymbol{x}\right] \leq \|\boldsymbol{x}\|_G^2 - a_n k_0 \|\boldsymbol{x}\|_G^2 + a_n^2 \left\{k_1' \|\boldsymbol{x}\|_G^2 + c_0 + c_1 \|\boldsymbol{x}\|_G^2\right\} \tag{27}$$

$$E\left[\|\boldsymbol{x}(n+1)\|_G^2 \mid \boldsymbol{x}(n) = \boldsymbol{x}\right] \geq \|\boldsymbol{x}\|_G^2 - a_n k_0' \|\boldsymbol{x}\|_G^2 + a_n^2 \left\{k_1 \|\boldsymbol{x}\|_G^2 + d_0\right\} \tag{28}$$

We now consider the simple minded algorithm we proposed in (15) by assuming $a_n = \epsilon$ in the following derivations. Taking the expectation of both sides of the final inequalities (27) and (28) with respect to $\boldsymbol{x}(n)$ and letting $b_n = E[\|\boldsymbol{x}(n)\|_G^2]$ and $a_n = \epsilon$, we obtain,

$$\left\{1 - k_0'\epsilon + k_1\epsilon^2\right\} b_n + \epsilon^2 d_0 \leq b_{n+1} \leq \left\{1 - k_0\epsilon + (k_1' + c_1)\epsilon^2\right\} b_n + \epsilon^2 c_0 \tag{29}$$

By defining $\alpha_0 = 1 - k_0\epsilon + (k_1' + c_1)\epsilon^2$ and $\alpha_1 = 1 - k_0'\epsilon + k_1\epsilon^2$ we can write the lower and upper bounds on the non-negative sequence b_n as,

$$\alpha_1 b_n + \epsilon^2 d_0 \leq b_{n+1} \leq \alpha_0 b_n + \epsilon^2 c_0 \tag{30}$$

Therefore, non-negative sequence b_n is sandwiched between two sequences generated according to $b'_{n+1} = \alpha_0 b'_n + \epsilon^2 c_0$ and $b''_{n+1} = \alpha_1 b''_n + \epsilon^2 d_0$. Those two sequences converge to finite numbers if and only if ϵ is chosen such that $|\alpha_0| < 1$ and $|\alpha_1| < 1$.

We note that α_0 and α_1 are equal to 1 at $\epsilon = 0$. We also note that both α_0 and α_1 are locally decreasing as ϵ increases, since $d\alpha_0/d\epsilon|_{\epsilon=0} = -k_0 < 0$ and $d\alpha_1/d\epsilon|_{\epsilon=0} = -k_0' < 0$. This means that we can choose ϵ small enough so that $|\alpha_0| < 1$ and $|\alpha_1| < 1$ in which case the sequences b'_n and b''_n converge and the limiting G-norm MSE, i.e., $\lim_{n\to\infty} b_n$, has finite lower and upper bounds. From the sandwich theorem, we have

$$\frac{\epsilon^2 d_0}{1 - \alpha_1} \leq \lim_{n\to\infty} b_n \leq \frac{\epsilon^2 c_0}{1 - \alpha_0} \tag{31}$$

We can evaluate the lower and upper bounds in the extreme case when $\epsilon \to 0$ as,

$$\lim_{\epsilon \to 0} \frac{\epsilon^2 d_0}{1 - \alpha_1} = \lim_{\epsilon \to 0} \frac{\epsilon d_0}{k_0' - k_1 \epsilon} = 0 \qquad \text{and} \qquad \lim_{\epsilon \to 0} \frac{\epsilon^2 c_0}{1 - \alpha_0} = \lim_{\epsilon \to 0} \frac{\epsilon c_0}{k_0 - (k_1' + c_1)\epsilon} = 0 \quad (32)$$

Therefore, for arbitrarily small ϵ, both lower and upper bounds for b_n approach zero, implying that the limiting G-norm MSE goes to zero as well. In this case, Lemma 2 implies that the limiting MSE goes to zero and the SPCA (15) converges to the unique optimal power vector $\bar{\boldsymbol{p}}$. However, as ϵ approaches zero, α_0 and α_1 approach 1 which slows the convergence rate. Instead of choosing a very small ϵ, one should try to choose that value of ϵ which makes both α_0 and α_1 as close to zero as possible so as to increase the convergence rate. After fixing the value of ϵ we can use *averaging* to decrease both the lower and the upper bounds on the limiting MSE.

In order to solve for the value of ϵ that gives rise to acceptable values of α_0 and α_1 in terms of the convergence rate and lower and upper bounds on limiting MSE, one needs to know the constants k_0, k_0', k_1, k_1' and c_1 which depend on the global system parameters such as the eigenvalues of matrix $\boldsymbol{H}^{-1}\boldsymbol{A}$. This calculation is difficult because it requires knowledge of all correlation coefficients and channel gains of all users in the system, and is computationally very expensive as the number of users N increases.

In order to overcome this difficulty, we propose to use a coefficient sequence a_n that is a function of n. The following theorem (see [12] for a proof) states that if a_n satisfies two simple conditions then the resulting SPCA converges to the optimal power vector $\bar{\boldsymbol{p}}$ in the mean squared sense irrespective of other system parameters, as long as the deterministic power control problem is feasible.

Theorem 5 *If* $\sum_{n=1}^{\infty} a_n = \infty$ *and* $\sum_{n=1}^{\infty} a_n^2 < \infty$ *and if the deterministic power control problem is feasible then* $\lim_{n \to \infty} E\left[\|\boldsymbol{x}(n)\|^2\right] = 0$.

5 The Stochastic Power Control Algorithm (SPCA)

We note that $a_n = a/n$ satisfies conditions given in Theorem 5. In particular we will chose $a = 1$. Starting from (16) of Section 4, and inserting $a_n = 1/n$ and $v_i(n) = y_i^2(n)$ we get the final form of the iterative SPCA.

$$p_i(n+1) = \left(1 - \frac{(1+\gamma^*)}{n}\right) p_i(n) + \frac{\gamma^*}{h_{ii}} \frac{y_i^2(n)}{n} \quad (33)$$

The algorithm given in (33) is distributed in the sense that for a mobile to update its power level it has to know only its own channel gain to its assigned base station and its own matched filter output.

In a typical implementation of SPCA in a wireless cellular system, a mobile transmits information bits to its assigned base station and that base station measures and transmits back to the mobile the value of the matched filter output, y_i, belonging to the mobile. Transmission of y_i would require possibly more than one bit depending on the number of levels to which y_i is quantized. If the base station transmits y_i at the end of each bit interval, an unacceptable level of traffic would be created for the power control purposes. To overcome this problem we propose an averaging scheme in which the base station observes the matched filter output for L bits, and averages the values of y_i^2 over the L bit interval. Every L bits, called a power control iteration cycle, the base station transmits the averaged value of y_i^2 which the mobile uses to adjust its transmitted power.

Because the power vector does not change during the L bit iteration cycle, the output

of the ith matched filter in the lth bit of the nth cycle can be written as,

$$y_i^2(n,l) = \sum_{j=1}^{N} \Gamma_{ij}^2 h_{ij} p_j(n) + w_i(n,l) \tag{34}$$

The averaged energy $\tilde{y}_i^2$ calculated over the L bit interval is

$$\tilde{y}_i^2 = \frac{1}{L} \sum_{l=1}^{L} y_i^2(n,l) = \sum_{j}^{N} \Gamma_{ij}^2 h_{ij} p_j(n) + \frac{1}{L} \sum_{l=1}^{L} w_i(n,l) \tag{35}$$

where $w_i(n,l)$ denotes the noise term at the end of the lth bit in the nth iteration cycle. Using $b_{j,l}$ and $n_{i,l}$ to denote the bit transmitted by user j and the AWGN sample at the output of the ith matched filter in the lth subinterval of nth iteration, we observe that: (i)$b_{j,l}$ and $b_{j,k}$ are independent, (ii) $n_{i,l}$ and $n_{i,k}$ are independent, (iii) the transmitted bits and channel AWGN noise are independent in the bit interval and across the bits. From these conditions, it can be easily shown that Lemmas 4 and 5 are still valid with the new definition of the observation vector $\tilde{\boldsymbol{v}}(n) = [\ \tilde{y}_1^2\ \tilde{y}_2^2 \ldots\ \tilde{y}_N^2\ ]$. This implies that the convergence theorem holds using the averaged measurements. The modified power control algorithm becomes

$$p_i(n+1) = \left(1 - \frac{(1+\gamma^*)}{n}\right) p_i(n) + \frac{\gamma^*}{h_{ii}} \frac{1}{n} \left(\frac{1}{L} \sum_{l=1}^{L} y_i^2(n,l)\right) \tag{36}$$

6 Results and Simulations

In our simulations, we consider a single cell of a wireless cellular CDMA system and ignore the intercell interference. We assume that the mobiles are uniformly distributed on a disk of radius R_0 (chosen as $R_0 = 1000$ meters) around the base station. The propagation constant α is chosen to be 4 in the simulations. Power control iterations are always started with the zero power vector. The simulations are over $10,000$ bits. If averaging of L bits is implemented then the number of power control iterations is $10,000/L$.

First we investigate the performance of the SPCAs where the sequence a_n has a constant value for all n, i.e., $a_n = \epsilon$. We considered a situation where there are $N = 20$ mobiles in the system. We assumed a processing gain of $G = 150$ and SIR target of $\gamma^* = 2$ (≈ 3 dB), and an AWGN with standard deviation $\sigma = 0.1$. The normalized squared error (NSE) which we define as $\text{NSE}(n) = \|\boldsymbol{p}(n) - \bar{\boldsymbol{p}}\|^2 / \|\bar{\boldsymbol{p}}\|^2$ is plotted as a function of iteration index in Figure 1. Different curves in this graph show the performance of the SPCAs with different $a_n = \epsilon$ values and that of the SPCA with $a_n = 1/n$. Figures 2 and 3 show the same performance criteria when averaging is implemented with $L = 10$ and $L = 100$, respectively.

We observe the trade-off between the convergence rate and the value of the limiting NSE: when ϵ is large, α_0 and α_1 are smaller, and the convergence rate is fast, but the limiting NSE is larger. Therefore, for this case we observe an initial fast decrease in the NSE, but then oscillations around the limiting NSE; see the $\epsilon = 0.01$ curve in Figure 1. On the other hand, if ϵ is close to zero, then the limiting value of the NSE is smaller, but since α_0 and α_1 are close to 1 the convergence rate is very slow. In this case, we observe a slowly but steadily decreasing NSE with little oscillation; see $\epsilon = 0.0001$ curve in Figure 1. Also we observe from Figures 1, 2 and 3 that the performance of the SPCA with $a_n = \epsilon$ is almost the same as the performance of the SPCA which uses averaging over L bits with $a_n = L\epsilon$.

In order to show the convergence of the SIRs of the users to the target SIR, we ran the SPCA with $a_n = 1/n$ and plotted the SIRs of all the users as a function of the iteration index n. The graph is shown in Figure 4. In this experiment we chose $N = 20$, $G = 150$, $\sigma = 0.1$ and $\gamma^* = 4$ (≈ 6 dB). We did the same experiment for the SPCAs with the fixed $a_n = \epsilon$ sequence. Results are plotted in Figures 5 and 6 for the cases $\epsilon = 0.001$ and $\epsilon = 0.0001$, respectively. We observed that with $a_n = 1/n$, SIRs converge to the target SIR as expected. For fixed $a_n = \epsilon$, we observed the trade-off between the convergence rate and oscillations around the convergence point: as ϵ increases the convergence rate increases, but the SIR sequences oscillate around the target value with increasing amplitude.

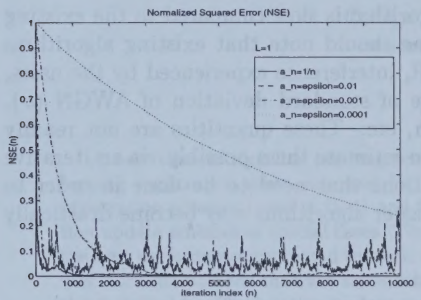


Figure 1: NSE versus n for the SPCAs with $a_n = 1/n$, and $a_n = \epsilon$. ($L = 1$)

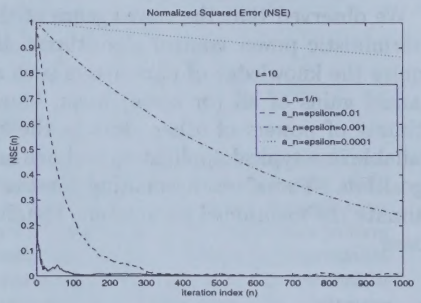


Figure 2: NSE versus n for the SPCAs with $a_n = 1/n$, and $a_n = \epsilon$. ($L = 10$)

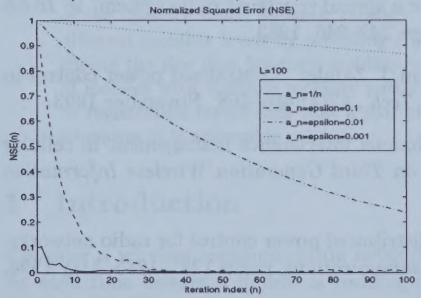


Figure 3: NSE versus n for the SPCsA with $a_n = 1/n$, and $a_n = \epsilon$. ($L = 100$)

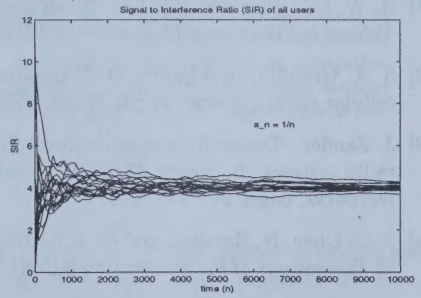


Figure 4: SIRs of all the users as a function of n. SPCA with $a_n = 1/n$. ($L = 1$)

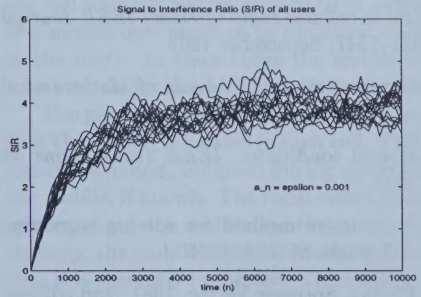


Figure 5: SIRs of all the users as a function of n. SPCA with $a_n = 0.001$. ($L = 1$)

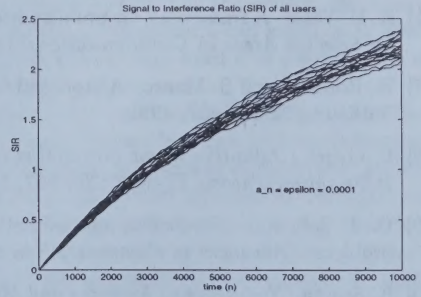


Figure 6: SIRs of all the users as a function of n. SPCA with $a_n = \epsilon = 0.0001$. ($L = 1$)

7 Conclusion

We proposed two classes of stochastic power control algorithms (SPCAs), where a fixed coefficient sequence was used in the first class and $a_n = 1/n$ was used in the second class. We investigated the conditions under which we can have lower and upper bounds on the limiting value of mean squared error (MSE) for the first class of SPCAs.

We showed that if $a_n = 1/n$ type of a coefficient sequence was used in the power control iterations then the limiting value of MSE went to zero, given that the deterministic power control problem was feasible. We also showed that a version of this SPCA which used averaging converged to the global optimum power vector if the deterministic power control algorithm was feasible.

We observed that the convergence of the algorithm is slow compared to the existing deterministic power control algorithms. But one should note that existing algorithms require the knowledge of parameters such as SIR, interference experienced by the users, channel gains of all (or some) users, knowledge of standard deviation of AWGN (σ), estimate of powers of other users in the system, etc. These quantities are not readily available in a typical application and one needs to estimate them possibly via an iterative algorithm. Therefore, accounting for the iterations that need to be done in order to estimate the mentioned parameters, seemingly faster algorithms may become drastically slower.

References

[1] R. W. Nettleton and H. Alavi. Power control for a spread spectrum radio system. In *IEEE Vehicular Technology Conference VTC-83*, pages 242–246, 1983.

[2] S. A. Grandhi, R. Vijayan, D. J. Goodman, and J. Zander. Centralized power control in cellular radio systems. *IEEE Trans. Vehicular Tech.*, 42(4):466–468, November 1993.

[3] J. Zander. Transmitter power control for co-channel interference management in cellular radio systems. In *Fourth WINLAB Workshop on Third Generation Wireless Information Networks*, pages 241–247, 1993.

[4] S. C. Chen, N. Bambos, and G. J. Pottie. On distributed power control for radio networks. In *Proceedings of the International Conf. on Communications*, pages 1281–1285, May 1994.

[5] G. J. Foschini and Z. Miljanic. A simple distributed autonomous power control algorithm and its convergence. *IEEE Trans. Vehicular Tech.*, 42(4):641–646, November 1993.

[6] R. D. Yates. A framework for uplink power control in cellular radio systems. *IEEE Journal on Selected Areas in Communications*, 13(7):1341–1347, September 1995.

[7] H. Robbins and S. Monro. A stochastic approximation method. *Annals of Mathematical Statistics*, 22:400–407, 1951.

[8] L. Gyorfi. Adaptive linear procedures under general conditions. *IEEE Transactions on Information Theory*, IT-30(2):262–267, March 1984.

[9] D. J. Sakrison. Stochastic approximation : A recursive method for solving regression problems. *Advances in Communication Systems 2*, pages 51–106, 1966.

[10] E. Seneta. *Non-negative Matrices and Markov Chains.* Springer Verlag, 1981. 2nd edition.

[11] D. D. Siljak. *Large-Scale Dynamic Systems : Stability and Structure.* North-Holland, 1978.

[12] S. Ulukus and R. D. Yates. Stochastic power control for cellular radio systems. Submitted to IEEE Transactions on Communications, September 1996.

LOCATION TRACKING MECHANISMS FOR OPTIMAL MOBILITY ADAPTATION *

Leandros Tassiulas and Farooq M. Anjum†
Electrical Engineering
University of Maryland
College Park, MD 20742
leandros@isr.umd.edu

Abstract

A new class of mobility tracking schemes are proposed where multiple layers of overlapping registration areas are used for location updates. This class includes the static registration schemes (used in IS-41 and GSM) as well as the distance based dynamic location update schemes as special cases. The latter has been shown in previous studies to have the best performance compared to static and timing based schemes, but its implementation was more complicated than the static schemes due to significant processing requirements in the mobile side. The location update method proposed here is amenable to an implementation with complexity of the level of the static registration area based strategy and with minimal processing requirements on the mobile side while its performance is of the level of the distance based dynamic strategy. Furthermore, our scheme supports adaptation to different mobility levels by providing registration areas of multiple sizes for the mobiles to choose the size that fits their mobility pattern. In the case of a heterogeneous population of mobiles with multiple mobility patterns we provide an algorithm for allocating mobiles to registration layers such that a pair of target paging rates and registration rates is met whenever it is achievable.

1 Introduction

Consider a wireless communication network with M cells and a large number of mobile users. At each time instant a mobile is residing in one of the M cells. If it is paged by the base of that cell, it will respond to the paging by a registration message and its current location will be identified by the system. Since the mobile is moving, the system in general does not know the cell where it resides at each time instant. During the operation of the system, the location of a mobile user needs to be identified at certain times (i.e. whenever a call needs to be placed to the user). At these times the mobile is paged in different cells until it is paged in the cell where it resides in which case its location is identified by the system.

The paging controller determines the paging strategy for each mobile based on the likelihood that the mobile is located in each cell. This is determined in turn based on location information about the mobile, obtained during *registration*, as well as the statistics of the mobility pattern of the mobile, if known. The registration strategy is a mechanism that provides information to the system about the mobile locations. At certain time instances, determined by the registration strategy, the mobile registers itself, that is it informs the base station of the cell where it resides, that it is currently in the cell. There are several different approaches to the registration process.

*This paper appeared in part in the IEEE Intl. Conference on Universal Personal Communications

†This research was supported in part by NSF under grant NCR-9406415, a CAREER award NCR-9502614, by the AFOSR under grant 95-1-0061 and ATIRP Consortium sponsored by the U.S. Army Research Laboratory under the Federated Laboratory Program, Cooperative Agreement DAAL01-96-2-0002.

The registration strategy currently in use in cellular telephony is based on the concept of *disjoint registration areas* [8, 9]. The collection of all cells in the system is partitioned in a number of registration areas. Each cell broadcasts the ID of the registration area to which it belongs. Whenever a mobile crosses the boundary of two cells belonging to different registration areas, it registers itself to the cell that it entered. Hence at every time instance the system is aware of the registration area where the mobile belongs to, therefore it need not be paged in any cell outside of the most recent registration area. If in addition information about the statistics of the mobility pattern is available, the paging controller might be able to estimate the probability that the mobile is located in each one of the cells of the registration area and subsequently use this information to increase the efficiency of the paging.

A number of dynamic registration strategies have been proposed recently in the literature, for increasing the efficiency of the mobility management process. One of the most popular is distance based registration [3, 6, 7]. In distance based registration, a mobile registers itself whenever it enters a cell such that its distance from the cell where the mobile was registered last time is greater than or equal to $\bar{d}$. Hence at any time instance, the paging controller need only page the mobile in some or all cells the distance of which is less than or equal to $\bar{d}$ from the cell where the mobile was registered for last time. As in the case of the strategy based on registration areas, information about the statistics of the mobility pattern can be used, if available.

One of the main considerations regarding a mobility management strategy is the signaling complexity required for its implementation. There are two components in it. The signaling in the wireless channel, that is paging and registration messages, and signaling in the wired network, that is message exchange for location database access and updates [4, 5, 6]. The focus of this paper is on the wireless part, the performance of which is characterized by the rates of registration messages in the uplink channel and paging messages in the downlink channel. Among the many strategies [1, 2, 3, 7, 10, 13] that have been considered recently the dynamic distance based strategy appears to have the best performance. Nevertheless, its implementation is not simple and more importantly it is computationally intensive on the mobile side, something that makes it unsuitable for implementation. The registration scheme proposed here is static and simple to implement and can achieve performance identical with the distance based dynamic strategy or very close to that with minimal additional overhead as compared to the static strategy with disjoint registration areas. The other feature of the registration scheme is that it provides flexibility for adaptation to the mobility profile of the user, by allowing the user to choose among registration areas of various sizes as we will see in section 5.

The following assumptions are made throughout the paper. The wireless system under consideration is a cellular network represented by a regular hexagonal lattice as shown in figure 1. Let $X(t)$ be the cell where the user resides at time t. The mobility profile of a user is represented by the *location process* $X = \{X(t), t \geq 0\}$. This is a piece-wise constant process with state space the set of cells of the system. The state transitions correspond to crossings of cell boundaries. The location process is a homogenous Markov chain. The probabilities of transition from a cell i to each one of its six neighbors is equal to $1/6$. The residence time in a cell is exponential with rate v, that is the average residence time in a cell is $1/v$. The rate v reflects the speed of the mobile. Each mobile is receiving calls, therefore it needs to be located, according to a Poisson process of rate λ. The call/mobility type of the mobile is characterized by the pair (λ, v).

2 The overlapping registration areas structure

The type of registration/paging mechanism that we propose relies on a structure of registration areas of the following type. A collection $\mathcal{C} = \{S_i : i = 1, .., J\}$ of J subsets of cells (*registration areas*) is specified. This collection covers all cells in the system in the sense that every cell is

included in at least one set of $\mathcal{C}$. The cells in each set S_i are contiguous or in other words no set S_i includes disjoint regions of cells. The sets S_i might be arbitrarily overlapping otherwise.

Each mobile is associated (*registered*) with exactly one registration area at each time t. This area always contains the cell where the mobile currently resides. As soon as the mobile enters a cell that does not belong to the registration area the mobile was most recently registered to, it needs to be registered again to one of the registration areas where the new cell belongs to. In the rest of the paper we will consider registration areas that consist of a collection of cells in a regular hexagonal arrangement, as shown in figure 1. The *size* d of such a registration area will be taken to be the number of cells in one side of the hexagon minus one.

The type of registration strategies described above is fairly general. If the sets in $\mathcal{C}$ are disjoint then the registration strategy coincides with the classical one based on disjoint registration areas. If there is one set S_i associated with each cell i such that S_i is centered in i and its size is equal to d, then the dynamic distance based registration strategy can be implemented in this manner.

A mobile registers itself informing the system about its location when it moves out of its current registration area. In this case the mobile may select to register at any of the registration areas that the cell where it resides at that time belongs to. The registration rate g_r, in number of registration messages per cell boundary crossings, depends on the registration scheme and is a measure of its performance. In certain cases we will need to refer to the registration rate expressed in registration messages per time unit, that is $\hat{g}_r = v^{-1} g_r$.

The mobile is paged whenever there is a call directed to it. The paging rate g_p, in paging messages per time unit, depends both on the rate with which a mobile is receiving calls and on the paging mechanism. In the simplest registration strategy, which is the one currently employed in cellular telephony, the mobile is paged in all cells of its registration area S_i at the time the call is placed. Hence the number of paging messages generated is equal to $|S_i|$, the number of cells in S_i. If p_i is the steady state probability that the mobile is registered in area S_i and assuming Poisson call generation process, the paging rate in pagings per location request is

$$g_p = \sum_{i=1}^{M} p_i |S_i| \tag{1}$$

If all the registration areas are of equal size $|S|$, then the paging rate becomes

$$g_p = |S| \tag{2}$$

The paging rate in paging messages per time unit is $\hat{g}_p = \lambda g_p$.

Several methods for paging are under investigation currently [11], which try to reduce the paging rate by employing sequential paging instead of blanket paging to all the cells of the registration area. If any of these methods is employed, the paging rate is less than that in equation (1). In this paper, it is assumed that a mobile is paged simultaneously in all cells where it may reside. Therefore, the paging rate is given by (1) or (2). If more sophisticated paging strategies are used, then the paging rate may be equal to a fraction of the registration area size. In that case also the trade-off between g_r and g_p remains qualitatively the same.

3 Single layer overlapping registration areas

In a single layer registration system, all the registration areas have the same size and they are regular hexagons. A single layer system is specified by the common size of the registration areas and their locations. Note that given the cell in the center of the registration area and its size, the registration area is fully specified. Hence a single layer system is specified by giving the centers of all its registration areas. For the latter purpose a 60^o degree coordinate system is adopted, as depicted in figure 1. One registration area is always centered in the cell with coordinates

(0,0). The neighborhood structure of the layer is specified by the registration areas which are either overlapping or tangential to the one centered in the cell with coordinates $(0,0)$. This is done by the collection of vectors that specify the displacement of the centers of the registration areas from the cell $(0,0)$.

Consider for instance the single layer system consisting of disjoint registration areas of sizes $d = 4$ as shown in figure 1. Each registration area is tangential to 6 others; the displacement of the centers of the registration areas tangential to the one centered in cell $(0,0)$ are (d,d), $(-d,2d)$, $(-2d,d)$, $(-d,-d)$, $(d,-2d)$, $(2d,-d)$. The collection of the centers of all registration areas of the layer is then given as $\{(dk_1,dk_1)+(2dk_2,-dk_2)+(dk_3,-2dk_3)+(-dk_4,-dk_4)+(-2dk_5,dk_5)+(-dk_6,2dk_6)$: $k_1,k_2,k_3,k_4,k_5,k_6 = 0,1,2,\cdots\}$. On the other hand consider the single layer system containing one registration area of size d for each cell of the system, centered in the cell. The collection of the centers of all registration areas in this case is $\{(k,m) : k,m = 0,\pm1,\pm2,..\}$. Note that in this case a cell i belongs to all registration areas centered in cells within the registration area of i. Hence, the number of possible distinct registration areas equals the number of cells in the registration area. Whenever a mobile moves out of its registration area, it registers again to the registration area centered in the new cell.

In our performance study we will consider two additional single layer systems. One in which each registration area overlaps with three others and another where each registration area overlaps with six others. Examples of each one of the above cases with registration areas of size $d = 5$ is depicted in figure 2. The first single layer system will be referred to as the 3-overlapping system; the centers of the registration areas overlapping with the one centered in $(0,0)$ are in $(d,0)$, $(-d,d)$, $(0,-d)$ and the collection of the centers of all registration areas of the layer is $\{(dk_1,0)+(0,-dk_2)+(-dk_3,dk_3) : k_1,k_2,k_3 = 0,1,2,..\}$. In this system most of the cells belong to two registration areas. The second single layer system will be referred to as the 6-overlapping system. The centers of the registration areas overlapping with the one centered at $(0,0)$ are at $(d,0)$, $(d,-d)$, $(0,-d)$, $(-d,0)$, $(-d,d)$, $(0,d)$. The collection of the centers of all registration areas of the layer is $\{(dk_1,0)+(dk_2,-dk_2)+(0,-dk_3)+(-dk_4,0)+(-dk_5,dk_5)+(0,dk_6) : k_1,k_2,k_3,k_4,k_5,k_6 = 0,1,2,..\}$.

The performance of the system is characterized by the pair (g_p,g_r) of registration and paging rates respectively. In a single layer system the paging rate is a function of the size of the registration areas only and not of the relative locations of the areas or the registration scheme. If the size of the area is d, then the paging rate is

$$g_p = 1 + 3d(d+1) \tag{3}$$

In the following the discussion focusses on the analysis of the registration rate.

3.1 Evaluation of the registration rate

Let $\{t_i^r\}_{i=1}^{\infty}$ be the collection of all times at which the mobile registers itself. In stationary operation the registration rate $\hat{g}_r$ is

$$\hat{g}_r = \frac{1}{E[t_{i+1}^r - t_i^r]},$$

that is the inverse of the expected time between two successive crossings of the registration area boundary. Normalizing by the average cell residence time, the registration rate in registration messages per cell crossing is

$$g_r = \frac{v}{E[t_{i+1}^r - t_i^r]}, \tag{4}$$

Lets denote by E_c the set of all possible *entry* cells of the registration area, that is those cells which might be visited first when the user enters the registration area. In the case of disjoint registration areas for example, E_c contains all cells in the boundary of the registration region.

In the case of one registration area centered in each cell, E_c contains only the center cell of each registration area. The sets E_c for the 3-overlapping and 6-overlapping systems are illustrated in figure 3. Let $p_e(i)$ be the probability that a mobile will enter the registration area in cell i and $T(i)$ be the expected number of cell crossings until the mobile moves out of the registration area, from the time it starts in cell i. Then the registration rate can be expressed as

$$g_r = \frac{1}{\sum_{i \in E_c} p_e(i) T(i)}. \tag{5}$$

It is fairly intuitive (and can be easily shown) that the closer a cell i is to the center of the registration area, the higher $T(i)$ will be and in fact $T(i)$ is maximum when i is the center cell. Hence the registration rate will be smaller for registration schemes that result in sets of possible starting cells which are more concentrated around the center cell.

In figure 4 we see plots of the pair (g_p, g_r). Curves corresponding to different single layer systems are depicted. The points of each curve are obtained for different registration area sizes ranging from 1 to 10. As is expected, the strategy with one registration area per cell has the best performance. Note also that the 6-overlap system as well as the 3-overlap system have performance curves very close to that of the one registration area per cell system. This is achieved while each cell belongs to 2 or 3 registration areas respectively, and independently of the size of the registration areas compared to $1 + 3d(d+1)$ registration areas containing each cell for the latter strategy.

The loading of the system in terms of registration and paging messages per time unit $\hat{g}_p$ and $\hat{g}_r$ respectively depends on the mobility of the mobile. The performance curves can be normalized by the parameters λ and v each time to get the performance curve for a specific mobility type. Given the available capacities, or equivalently desirable paging and registration rates (g_p^o, g_r^o) in the downlink and uplink channels respectively, and the performance curves for the specific mobility type, we can determine which single layer strategies and corresponding registration area sizes are feasible under these requirements. For example, the strategies with performance points within the shaded rectangle, achieve paging and registration rates less than (g_p^o, g_r^o) respectively. Note that with disjoint registration areas the pair (g_p^o, g_r^o) is not achievable.

In general we may have a mix of mobiles with N possible call-mobility types characterized by the pairs (λ_n, v_n), $n = 1, \cdots, N$. Assume that M_n mobiles have the mobility pattern (λ_n, v_n). If there is a single layer of registration areas with normalized paging registration rate pair (g_p, g_r), then the corresponding rates per time unit will be

$$\hat{g}_r = \sum_{n=1}^{N} v_n^{-1} M_n g_r \quad , \quad \hat{g}_p = \sum_{n=1}^{N} \lambda_n M_n g_p.$$

In case of a traffic mix as above, a combination of registration areas may improve performance significantly.

4 Multilayer overlapping registration areas

Consider a multilayer registration scheme consisting of a number of overlayed single layer registration areas as the ones discussed in section 3. For simplicity and without loss of generality assume that all layers are of the same type as far as the overlapping is concerned and they differ only by the size of the registration areas. Assume that there are K layers with registration area sizes d_k, $k = 1, \cdots, K$ indexed in increasing order, that is $d_{k+1} > d_k$, $k = 1, \cdots, K-1$.

In a multilayer system, each mobile has the option to choose the registration areas layer that it will use to update its location registration. Let f_{nk} be the number of mobiles of type n allocated to use layer k for registrations. A feasible mobile allocation is any vector $F = \{f_{nk},\ n = 1, \cdots, N,\ k = 1, \cdots, K\}$ such that $f_{nk} \in Z_+$ and $\sum_{k=1}^{K} f_{nk} = M_n$, $n = 1, \cdots, N$. Let

$\mathcal{F}$ be the set of feasible mobile allocations. The paging-registration rates pair associated with a mobile allocation F is (g_p^F, g_r^F) where

$$g_r^F = \sum_{k=1}^{K} \sum_{n=1}^{N} v_n^{-1} f_{nk} g_r^k \quad , \quad g_p^F = \sum_{k=1}^{K} \sum_{n=1}^{N} \lambda_n f_{nk} g_p^k$$

The performance of a mobile allocation is characterized by its paging-registration rate pair, (g_p^F, g_r^F). Consider the set

$$G = ((g_p^F, g_r^F) : \ F \in \mathcal{F})$$

of all achievable paging-registration rate pairs by feasible allocations of mobiles to layers. The set G is called the performance region of mobile allocations in the following.

A location tracking scheme should select the allocation of mobiles to layers such that the resulting paging-registration rate pair is optimal for the performance criterion that is used for the system. In the following we show that there is a very limited set of performance points worth consideration in optimizing any performance measure of practical significance and a fairly restricted set of associated allocations.

In doing so we expand the set of allocations to include fractional allocations of a mobile to different layers, that is f_{nk} might be an arbitrary nonnegative real number. A feasible allocation is any vector $F = \{f_{nk}, \ n = 1, \cdots, N, \ k = 1, \cdots, K\}$ such that $f_{nk} \in R_+$ and $\sum_{k=1}^{K} f_{nk} = M_n, \ n = 1, \cdots, N$. Denote by $\tilde{G}$ the set of achievable paging-registration rate pairs under fractional allocations. It is easy to show that $\tilde{G}$ is the convex hull of G and it is also closed. Also, if the numbers of mobiles $M_n, \ n = 1, \cdots, N$ are large, any point in $\tilde{G}$ is closely approximated by some point in G.

A paging-registration pair g^1 *dominates* another pair g^2 if and only if

$$g^1 \leq g^2$$

where the inequality is meant to be component-wise. A mobile allocation F_1 *dominates* an allocation F_2 if and only if g^{F1} dominates g^{F2}.

Since $\tilde{G}$ is a bounded, closed convex set, there is a subset G_B of $\tilde{G}$ with paging-registration pairs that have the following properties:

- Every element of G_B is dominated only by itself.
- Every element in $\tilde{G}$ is dominated by some element in G_B.

Clearly for any performance measure of practical interest, only allocations with paging-registration rate pairs in G_B are worth considering. In the following we characterize a set of allocations with performance measures which achieve all the performance pairs in G_B.

Consider allocations where all mobiles of the same type are assigned to the same layer. These allocations F are specified alternatively by a corresponding layer vector $l = (l_n, \ n = 1, \cdots, N)$ where l_n is the layer to which all mobiles of type n are allocated to.

Consider the sequence of allocations F^i, with layer vector $l^i, \ i = 1, \cdots, N(K-1)$ which are as follows

1) $l_n^1 = 1, \quad n = 1, \cdots, N$

2) l^{i+1} is obtained from l^i as follows:

a)For each type n such that $l_n^i < K$ compute the ratio

$$\frac{v_n^{-1}(g_r^{l_n^i} - g_r^{l_n^i+1})}{\lambda_n(g_p^{l_n^i+1} - g_p^{l_n^i})}. \tag{6}$$

b) Select the type n_o such that $l_{n_o}^i < K$ and for which the ratio given by (6) is maximum.

c) Let $l_{n_o}^{i+1} = l_{n_o}^i + 1$ and $l_n^{i+1} = l_n^i, \ n \neq n_o$

Consider the set of allocations $\mathcal{F}_B$

$$\mathcal{F}_B = \{F: \ F = \alpha F^i + (1-\alpha)F^{i+1}, \ 0 \leq \alpha \leq 1, \ i = 1, \cdots, N(K-1)-1\}$$

This set contains all the important allocations in the sense described in the following theorem.

Theorem 1 *For any allocation $F \in \mathcal{F}$, there is some allocation $F' \in \mathcal{F}_B$ such that*

$$g^{F'} \leq g^F.$$

The above theorem follows from the convexity of the curve of achievable paging-registration rate pairs, for different registration area sizes. An extensive related study and a rigorous treatment of the properties of the algorithm is available in [12]. The question of feasibility of a paging-registration rate vector can be answered by considering only allocations in F_B and it is a fairly simple task given that F_B is a one-parameter family of allocations with performance space that is a monotonic curve.

Comparative evaluation of the performance of the single layer and the multiple layer strategies are shown in figure 5. For the multi-layer strategy we assume 10 layers having different registration area sizes ranging from 1 to 10. 8 different types of mobiles with different pairs of (λ_n, v_n) are assumed. Only one mobile is assumed to be in each type. The curve on the top represents the performance when all mobiles are assigned to the same layer, while the curve in the bottom is the lower boundary of the region of achievable rates. From repeated simulations it has been found that, the wider the disparities between the different types, the wider the performance gap becomes between the single layer and the multilayer case.

5 Implementation issues in the base and the mobile

The multilayer, overlayed registration areas scheme is amenable to a simple implementation along the lines of the implementation of the mobility management scheme in IS-41 and GSM. Recall that in the latter schemes each cell broadcasts periodically the ID of the unique registration area where it belongs. The mobile need only remember the area where it was registered for last time and constantly detect the registration area ID of its current cell. As soon as it detects that the ID is different than the one in its memory it updates its registration to the area where the current cell belongs to.

In our scheme each cell needs to broadcast all the registration areas where the cell belongs to in each layer. The information can be organized in a frame that is transmitted periodically as shown in figure 6. The registration area IDs of each layer are grouped together and the frame consists of the ID groups of all layers. Each mobile need only remember the layer where it is registered and the registration area in the layer. It should constantly monitor the group of registration area ID's where its current cell belongs to. If the registration area in its memory is not included in the ID group, then the mobile needs to update its registration. Note that in this case there are multiple options of registration areas in each layer. The one with center closest to the cell should be selected in each layer as was explained earlier. This information should be included in the frame either explicitly or implicitly. For instance each cell may place first in the ID group of each layer, the ID of the registration area that all the mobiles that register in the cell should select.

References

[1] I.F. Akyildiz and J.S.M. Ho. Dynamic mobile user location update for wireless PCS networks. *Wireless Networks*, 1:187–196, 1995.

[2] A. Bar-Noy and I. Kessler. Tracking mobile users in wireless networks. *IEEE trans. Inf. Theory*, 39:1877–1886, 1993.

[3] A. Bar-Noy, I. Kessler, and M. Sidi. Mobile users: To update or not to update. *Wireless Networks*, 1:175–185, 1995.

[4] R. Jain and Y.B. Lin. An auxiliary user location strategy employing forward pointers to reduce network impacts of pcs. *Wireless Networks*, 1:197–210, 1995.

[5] R. Jain, Y.B. Lin, C. Lo, and S. Mohan. A caching strategy to reduce network impact of pcs. *IEEE journal select. areas commun.*, 12:1434–1444, 1994.

[6] Y.B. Lin. Determining the user locations for personal communications services networks. *IEEE trans. Veh. Tech.*, 43:466–473, 1994.

[7] U. Madhow, M.L. Honig, and K. Steiglitz. Optimization of wireless resources for personal communications mobility tracking. *IEEE Transactions on Networking*, 3(6):698–707, Dec. 1995.

[8] S. Mohan and R. Jain. Two user location strategies for personal communications services. *IEEE Personal Communications*, pages 42–50, 1994.

[9] M. Mouly and M-B. Pautet. *The GSM System for mobile communications.* Palaiseau, France, 1992.

[10] S. Okasaka, S. Onoe, S. Yasuda, and A. Maebara. A new location updating method for digital cellular systems. page 345. Proc. IEEE 41st Vehicular Tech. Conf., May 1991.

[11] C. Rose and R. Yates. Minimizing the average cost of paging under delay constraints. *ACM Journal of Wireless Networks*, 1(2):211–219, 1995.

[12] Leandros Tassiulas and Farooq M. Anjum. *Efficient location tracking of mobile users:Tech. Report.* ISR,Univ of Maryland.

[13] H. Xie, S. Tabbane, and D. Goodman. Dynamic location area management and performance analysis. pages 536–539. Proc. IEEE VTC '93, 1993.

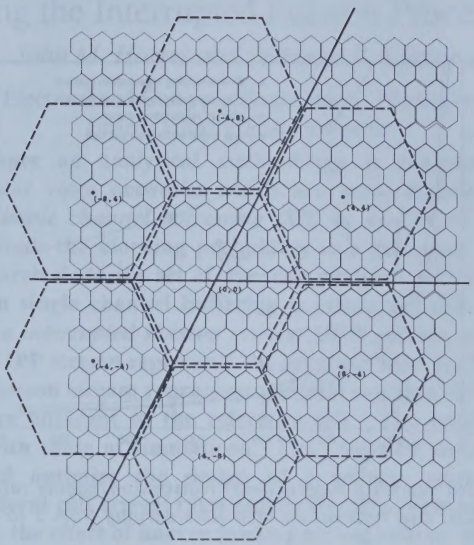


Figure 1: *A regular hexagonal cellular lattice is depicted. The 60° coordinate system is used. The registration areas are sets of cells with regular hexagonal shape.*

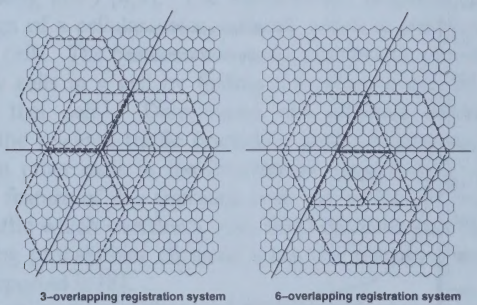


Figure 2: *Different single layer registration systems*

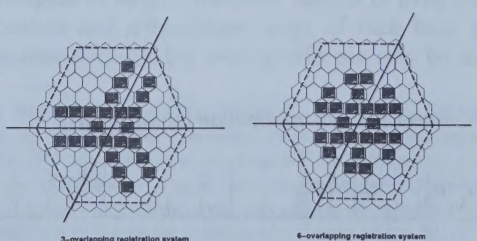


Figure 3: *Possible entry cells for the 3- and 6-overlapping registration system*

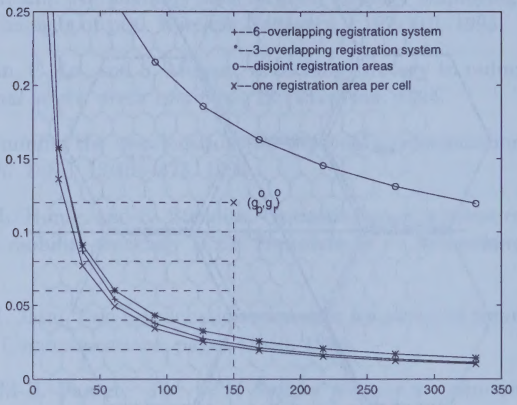


Figure 4: *The normalized (mobility independent) registration/paging rate pair curves for different registration strategies, and registration area sizes varying from 1 to 10*

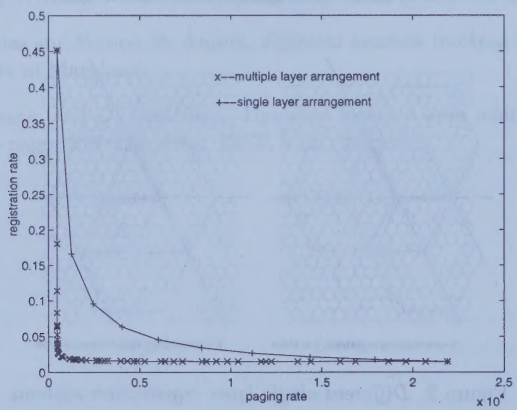


Figure 5: *Performance of the multilayer and single layer strategies for the 3-overlapping arrangement*

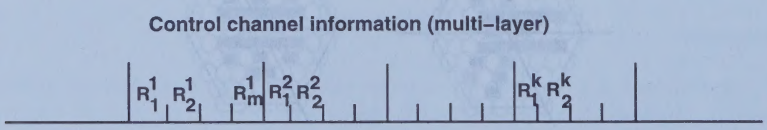


Figure 6:

Performance Analysis of Cellular Hybrid Channel Allocation using the Interrupted Poisson Process*

John M. Harris and Srikanta P. Kumar

Department of Electrical and Computer Engineering, Northwestern University
{jharris,kumars}@ece.nwu.edu

Abstract: In this paper an analytical methodology is proposed for assessing the performance of cellular voice networks which are using *Hybrid Channel Allocation (HCA)*, with either *simple channel borrowing (SB)* or *maximum packing (MP)*. The approach presented finds the blocking probability in a reference cell by modeling the composite stream of arrivals to the set of dynamic channels which are available in the reference cell. When single channel buffering is employed, this composite stream is approximated with an *interrupted Poisson process (IPP)* stream and a Poisson stream. In this approach, the IPP stream represents the overflow from the fixed channels in the reference cell. The Poisson stream represents the total overflow from the fixed channels in the cells which are adjacent to the reference cell. The results of this approach compare favorably with those of simulations. The proposed methodology can also be used to develop and optimize the design of a cellular voice network which is experiencing non-uniform call arrival rates among its cells. Few analytical models exist to date which capture the effect of uneven loading among cells on cellular networks.

I. INTRODUCTION

This paper presents a mathematical model which does performance analysis of cellular networks which are using *Hybrid Channel Allocation (HCA)* with *simple channel borrowing (SB)* or *maximum packing (MP)* [4,3]. The methodology proposed can be used to develop and optimize the design of a cellular voice network which is experiencing non-uniform call arrival rates among its cells. The performance analysis technique proposed finds the blocking probability of the new calls in a cell by finding the effective offered load to the dynamic channels available in that cell.[1] Determining this effective offered load involves the aggregation of all of the relevant traffic (including traffic originating in the given cell and overflow streams from other cells in the interference region) in a way that accounts for channel reuse. A key feature of this approach is its use of the *interrupted Poisson process (IPP)* to model the traffic which overflows from fixed channels into the dynamic channels in each cell. The blocking probabilities obtained using this technique compare favorably with those of a simulation reported in [8].

There are a number of simulation studies of HCA in the presence of uneven loading among cells [2,6,8]. However, few papers use mathematical analysis to examine the uneven loading scenario.[2] The analytical approach presented here can handle uneven traffic among cells and varying ratios of fixed to dynamic channels among cells. The methodology presented could help the system designer to select the optimal number of fixed channels for each cell as well as the optimal location and transmission range of each base station.[3] In contrast, methodologies which assume even loading among cells can only be used to select a system

* This work was partially supported by both the Daniel E. Noble IEEE Vehicular Technology Society/ Motorola Fellowship and by the Armed Forces Communications & Electronics Assoc. (AFCEA) Fellowship.

[1] The first model proposed in [17] also considered using the IPP for performance analysis of HCA. However, the model in [17] only uses the traffic within a single reuse cluster to predict the blocking probability.

[2] Of those papers which do consider this problem, most only address the one dimensional case [2,6,8,12].

[3] If the call arrival rate in each cell is known for a variety of cell sizes and locations, this approach can predict (and thus compare) the performance of the cellular network for each of the cell size/location scenarios.

wide fixed-to-dynamic channel ratio. HCA in cellular networks is introduced in section II. The intuitive overview of the model in section III is followed by the model itself in the section IV. Sections V, VI, & VII present numerical results, future work and conclusions.

II. Background and Terminology

The *cochannel interference constraint (CIC)* requires that the same channel not be used in any two cells which are closer than some distance R of each other.[4] In this model, R is chosen such that the same channel can't be used simultaneously in any two adjacent cells. This constraint is referred to as the *single cell buffering constraint*.[5] In *fixed channel allocation (FCA)*, the channels are first partitioned into the three groups, A, B, and C [8]. Each cell is then assigned one of the three cell types in the pattern depicted in Fig. 2. The three groups, A, B, and C, can contain different numbers of channels and still satisfy the CICs because no two cells of the same type are adjacent in Fig. 2.

Fig. 1: *Reuse Cluster*

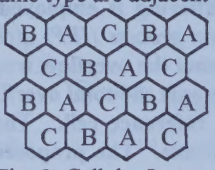


Fig. 2: Cellular Layout

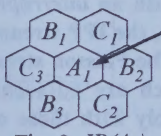


Fig. 3: $IR(A_1)$

A dynamic channel can be used in any cell where the CIC is satisfied. In *Dynamic Channel Allocation (DCA)*, all of the channels are classified as dynamic [1]. DCA and FCA are special cases of HCA. Some notation is introduced next. In HCA, the channels are first divided into a fixed group and a dynamic group. The channels in the fixed group are allocated according to FCA. When a new call is initiated in a cell, it is assigned a fixed channel, if one is available in the cell. If no fixed channels which are assigned to the cell are available, then a dynamic channel is assigned to the call according to a DCA scheme [3,4,5]. The stream of calls in each cell which do not find any fixed channels available in a cell is referred to as the *overflow stream* from the fixed channels in that cell. If a call in this overflow stream can not be allocated a dynamic channel by the DCA scheme, then it is *blocked from the system.*

A cell of interest is referred to as the *reference cell*. The model predicts the blocking probability of this cell. In this paper, cell A_1 is taken to be the reference cell. The *interference region (IR)* of the cell A_1 is the set of cells whose transmissions interfere with the cell A_1, i.e. $IR(A_1)=\{A_1, B_1, B_2, B_3, C_1, C_2, C_3\}$. See Fig. 3. Let $BC = \{B_1, B_2, B_3, C_1, C_2, C_3\}$. A channel is said to be *locked* in a cell if a new call to that cell can not use the channel because of the CIC. The channels locked in cell A_1 are those which are in use in at least one cell in $IR(A_1)$.

Let F and D be the set of fixed and dynamic channels respectively and let $F(x)$ be the set of fixed channels assigned to some cell x or cell type x.[6] Thus, the total set of channels available to the cellular service provider is $\{D \cup F\}$ where $F=\{F(A) \cup F(B) \cup F(C)\}$. The DCA schemes, MP and SB, are described in sections II A and B. The model proposed here

---

[4] Both *time division multiple access (TDMA)* and *frequency division multiple access (FDMA)* may be modeled using this algorithm. In TDMA, the time slot may be thought of as being a channel.

[5] This model can be adapted to model the multiple cell buffering scenario. In double cell buffering bc is the maximum of the number offered by six different seven cell reuse clusters for the MP case and the aggregation of six streams each steam being the maximum of the number offered by three different cells is bc for SB.

[6] If a channel has been permanently assigned for use in a specific cell, then it is said to be *locally fixed* in that cell. A channel is *locally dynamic* in cell A_1 if it is not locally fixed in any of the cells in the $IR(A_1)$. Thus, a channel may be locally fixed in one IR and yet still be locally dynamic in another. By unevenly partitioning the fixed channels, more channels can be reserved for use in the cells which are experiencing higher loads. This uneven allocation of fixed channels among cells can be accounted for in the methodology presented here, and thus can be included in the design procedure. However, for illustration, $F(B_i)=F(B)$ for $i=1,2,3$ here.

assumes that either SB or MP is being employed. As discussed in Section VI, the methodology proposed here can be modified to model HCA with other DCA schemes.

A. Maximum Packing (MP): [4]

MP implements an arbitrary number of dynamic channel reassignments in order to connect each new call. MP is also referred to as complete sharing. In MP, a new call is connected if doing so does not violate the *Reuse Cluster Constraint (RCC)*. The RCC requires that the number of dynamic channels in use in each reuse cluster is always less than or equal to the total number of dynamic channels, $|D|$. The numbers of dynamic channels in use in reuse clusters which are outside of the $IR(A_1)$ are unaffected by the connection of a new call in cell A_1. Thus, after an arrival to the cell A_1, the MP/RCC only needs to consider the number of dynamic channels which are in use in each of the reuse clusters within $IR(A_1)$.

If the RCC will be satisfied after a new call is connected, then MP assumes that there exists a set of channel reassignment such that the new call can be allocated a dynamic channel without violating the CICs. An example where this is true is depicted in Figs. 4 and 5 for $F=\{1,4,7\}$ and $D=\{2,3,5,6,8,9\}$. The RCC is a necessary but not sufficient condition for satisfying the cochannel interference condition. This is because the RCC only considers how many channels are in use in each reuse cluster, not which channels are in use in each reuse cluster. In some "rare" examples, the channel usage in the cells outside of the $IR(A_1)$ can create a situation where the RCC is satisfied by a new call in the cell A_1 and yet it is not possible to reassign the dynamic channels such that the CIC is satisfied after the new call is connected.[7] See [7] for an example. Thus, MP is an upper bound on performance.

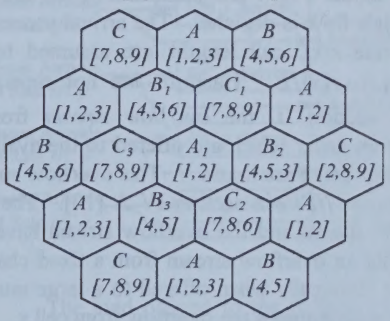


Fig. 4: Before New Call in A_1

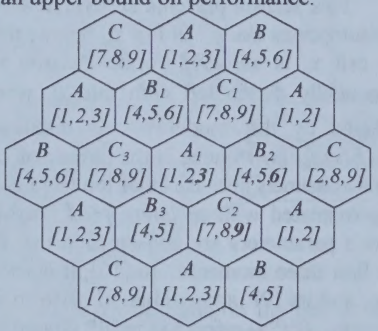


Fig. 5: After New Call in A_1

B. Simple Channel Borrowing (SB): [3]

SB is another DCA scheme. In SB, the dynamic channels are first "nominally" assigned to cells using a FCA-like policy. Each cell of type x are nominally assigned the dynamic channels in the set $N(x)$, where $D=N(A)\cup N(B)\cup N(C)$.[8] If a new call is placed in the cell A_1, and none of the dynamic channels nominally assigned to the cell A_1 is available, then the cell A_1 can *"borrow"* a dynamic channel from $N(B)$ or $N(C)$.[9] In this situation, SB chooses to borrow a dynamic channel which maximizes the minimum number of nominal channels available in the cells in BC. In other words, if $Avail(x)$ equals the number of available channels nominally assigned to cell x, then a channel borrowed for use in A_1 by SB is chosen such that the minimum of $(Avail(x): x\in BC)$ is maximized. In SB, when a call is

---

[7] i.e. Even after an arbitrary number of channel reassignments, the new call still can not be admitted.

[8] As is the case with fixed channels, it is also possible to vary the number of nominal channels assigned among cells of the same type. However, for simplicity, we assume that the same channels are nominally assigned to cells of the same type and that $N(A)$, $N(B)$ and $N(C)$ are disjoint.

[9] This is only if the channel is not in use in any of the cells adjacent to the borrowing cell.

terminated on a nominal channel, if a borrowed channel is being used in the same cell, then the nominal channel is reassigned so that the borrowed channel can be released.

We are assuming that the *Same Channel/Cell Type Property (SCCTP)* holds under SB. The SCCTP states that the same channels are used in cells of the same type. In other words, SCCTP states that for any two cells of the same type, the set of dynamic channels which are in use in one cell is contained within the set of dynamic channels which are in use by the other cell. For example, if the SCCTP is true, and the cells B_1, B_2, and B_3 are each using two, two, and three dynamic channels respectively, then the same two dynamic channels are known to be in use in the cells B_1, B_2, and B_3.

In Appendix A, an example is presented where the SCCTP holds only if SB is employed. In this example, it is also observed that fewer channels are borrowed when SB is employed. Because the dynamic channels which are borrowed are chosen such that *Avail(x)* is maximized, the SCCTP tends to be satisfied within each IR by the channels which are borrowed. Also, by inspection, when the blocking probability is low, few calls will be using borrowed channels. Thus, it is realistic to assume that SB has the SCCTP when the blocking probability is low. Given that the SCCTP holds, the number of dynamic channels which are in use in each cell in $IR(A_1)$ completely determines whether a new call can be connected in cell A_1. As discussed in Section VI, the approach presented here can be generalized to consider the effects of cells which are not in the IR of the cell A_1.

III. Model Overview

This section presents an overview of the model's flow and justifications for some of the assumptions made. In Fig. 6, below, the models flow is depicted. The arrival process to each cell x, is assumed to be Poisson with rate $r(x)$, call lengths are assumed to be exponentially distributed with rate μ, where $a(x) = r(x)/\mu$. Handoffs are not separately addressed by this model.[10] As discussed in section II, the *overflow stream* from a $M/M/|F(x)|(|F(x)|)$ queue is the stream of calls from cell x which are offered to the available dynamic channels because all of the fixed channels in cell x are in use.[11] This overflow stream is approximated with an *interrupted Poisson process (IPP)* in section IV A [10]. The IPP stream's parameters are chosen such that the IPP stream and the overflow stream have the same first three moments. In [10], it is shown that an overflow stream from a fixed channel group, and its IPP approximation perform almost identically when offered to a large number of servers. $IPP(x)$ refers to the IPP stream which approximates the overflow from cell x.

This approximation can also justified by noting that an IPP and an overflow steam are generated in a very similar fashion. An overflow stream is essentially a Poisson process when the queue is full, i.e. the fixed channels are all in use. When the queue is not full, no overflow occurs. Similarly, an IPP is composed of alternating *"on"* periods and *"off"* periods. During an *"on"* period, Poisson arrivals occur. Each *"on"* period has an exponentially distributed duration and is followed by an *"off"* period, during which no arrivals occur. *"Off"* periods for IPP also have exponentially distributed length. The only difference between an IPP and an overflow stream is that the overflow stream's *"off"* period is not exponential. Thus, the characteristics of an overflow stream are very similar to those of its IPP approximation. The motivation for the approximation of the overflow stream is that the interarrival distribution of an IPP has a much simpler analytical representation.

Thus, the overflow stream from the fixed channels in the each cell in the $IR(A_1)$ is approximated with an IPP. The load offered by cell x to the dynamic channels can be

---

[10] One may approximate the contribution of handoffs by increasing the Poisson arrival rates for each cell.

[11] The $M/M/|F(x)|(|F(x)|)$ loss queue represents the $|F(x)|$ fixed channels in cell x with Poisson arrivals with rate $r(x)$ and exponentially distributed service times (rate μ). When all servers are busy, arrivals are blocked.

characterized by either the interarrival distribution of *IPP(x)* or by the channel occupancy distribution of a $GI/M/\infty$ queue, where *GI=IPP(x)*.[12] The channel occupancy characterization is useful because it can be used to take channel reuse among the cells into account. For this reason, the distribution of the random variable which represents the offered load from cell *x* is calculated for each cell *x* in the $IR(A_1)$ using a $GI/M/\infty$ queue where *GI=IPP(x)* (in section IV B) [14]. The random variables, a_1, b_i, and c_i $(i=1,2,3)$, represent the number of dynamic channels which would be used by each of the corresponding cells in $IR(A_1)$ if the number of dynamic channels available is infinity, (load offered by each cell to *D*).

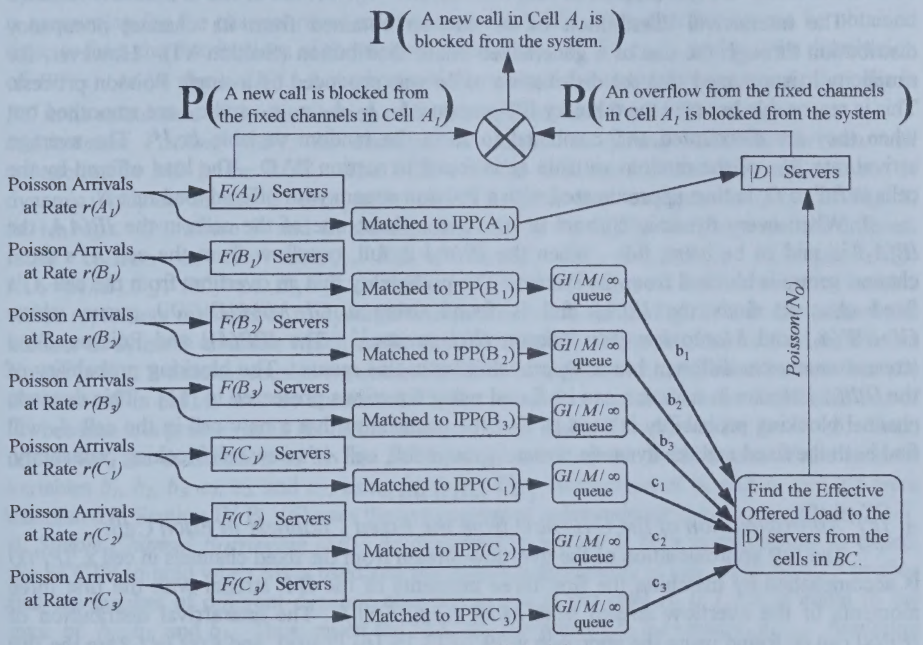


Fig. 6: Flow of Model

The load offered to the dynamic channels which are available in the cell A_1 may be viewed as the aggregate of several overflow streams. However, the IPP streams from the individual cells in *BC* must be discounted in this aggregation because of the channel reuse among the cells in *BC*. For example, if dynamic channel *i* is in use in the cells b_1, b_2, and b_3, then dynamic channel *i* would be locked only once in the cell A_1. This discounted aggregation of the overflow streams to the dynamic channels is done in section IV C using the following approach. The offered loads from each cell in *BC* (represented by b_1, b_2, b_3, c_1, c_2, and c_3) are combined into the random variable *bc* in a way which accounts for channel reuse among the cells. *P(bc=k)* is the probability that *k* dynamic channels would be locked in cell A_1 by the cells in *BC* if an infinite number of dynamic channels were available.[13] In other words, the random variable *bc* represents the total load offered by the cells in *BC* to the dynamic channels. The distribution of *bc* can be found from the distributions of b_1, b_2, b_3, c_1, c_2, and c_3. The method of discounted aggregation depends on the DCA scheme used, i.e. MP or SB.

In MP, the number of dynamic channels locked in cell A_1 is equal to the maximum number of dynamic channels in use in any pair of adjacent cells contained in *BC*,

---

[12] For example, Poisson (λ) arrivals with exponential (μ) call lengths have an offered traffic of (λ/μ) Erlangs.

[13] The number of dynamic channels is taken to be infinite here so that the "offered" load can be approximated.

$bc=\max(b_1+c_1,b_2+c_1,b_2+c_2,b_3+c_2,b_3+c_3,b_1+c_3)$. This occurs because the number of channels in use in each reuse cluster in Fig. 8 is locked in the cell A_1 (section II A). In contrast, if SB is employed, then the number of dynamic channels locked in the cell A_1 is equal to the maximum of the number of dynamic channels in use in cells B_1, B_2, and B_3 plus the maximum of the number of dynamic channels in use in the cells C_1, C_2, and C_3, $(bc=\max(b_1,b_2,b_3)+\max(c_1,c_2,c_3))$ (sections II A & IV C ii). Using these relationships, the distribution of bc can be found from the distributions of b_i *and* c_i for $i = 1, 2,$ and 3 (i.e. from the distribution of the $IPP(B_i)/M/\infty$ and $IPP(C_i)/M/\infty$ queues, $i=1,2,3$) (section IV C).

The interarrival distribution of bc can be obtained from its channel occupancy distribution through the use of a generalized phase distribution (Section VI). However, for simplicity it is assumed that the distribution of bc was produced by a single Poisson process. This is reasonable because the 6 bursty IPP streams, b_1, b_2, b_3, c_1, c_2, and c_3, are smoothed out when they are discounted and combined to form the random variable bc.[14] The average arrival rate, N_L, of the random variable bc is found in section IV D. The load offered by the cells in BC to D, is then approximated with a Poisson stream with offered load rate N_L.

When every dynamic channel is in use in at least one of the cells in the $IR(A_1)$, the $IR(A_1)$ is said to be being full. When the $IR(A_1)$ is full, overflow from the cell A_1's fixed channel group is blocked from the system. The probability that an overflow from the cell A_1's fixed channels finds the $IR(A_1)$ full is found using a $GI+M/M/|D|(|D|)$ queue where $GI=IPP(A_1)$ and *Markovian (M)*=Poisson (N_L) stream.[15] The $IPP(A_1)$ and Poisson (N_L) streams encounter different blocking probabilities in this queue. The blocking probability of the $IPP(A_1)$ stream in isolation can be found using equations presented in [11]. This dynamic channel blocking probability is used to find the probability that a new call in the cell A_1 will find both the fixed and the dynamic channel groups full, cell A_1's new call blocking probability.

IV. Model

A. IPP Approximation of the Overflow from the Fixed Channels in Each Cell:

The IPP approximation of the overflow stream from the fixed channels in cell x, $IPP(x)$ is accomplished by matching the first three moments of the IPP stream with the first three moments of the overflow stream from $M/M/|F(x)|(|F(x)|)$. The interarrival distribution of $IPP(x)$ can be found using the approach in [9,10,13,15,16].[16] $v_i(x)$, and $k_i(x)$ $i=1,2$ are the IPP parameters for the interarrival distribution, $E_x(t)$, of the $IPP(x)$. Use $a(x)$ and $F(x)$ to compute these parameters for each cell in $IR(A_1)$. Since the offered load, not the actual call arrival rate, is used to create the IPP interarrival distribution, $E_x(t)$ represents offered load.

$$E_x(t) = k_1(x)\left(1-e^{-v_1(x)t}\right)+k_2(x)\left(1-e^{-v_2(x)t}\right) \quad (1)$$

It is necessary to find the channel occupancy distribution of the load offered by each cell in $IR(A_1)$ to the dynamic channels. The number of calls which are *offered* by cell x to the dynamic channels is the number of dynamic channels which would be in use in cell x if an infinite number of dynamic channels were available (i.e., if it was possible to allocate a dynamic channel to every overflow call from the fixed channels in cell x). Thus, the number of

---

[14] An IPP stream is more bursty than a Poisson process. If the load in each cell is used in the determination of the number of fixed channels in each cell, then the overflow stream from the fixed channel group in each of the cells will have approximately the same rate. Thus, bc will more closely resemble a Poisson stream.

[15] For any x, $x \in BC$, an arrival to cell x is blocked from the system if the $IR(x)$ is full. This model only blocks calls from the cells in BC when the $IR(A_1)$ is full. Section VI proposes a modification which takes this into account.

[16] Equation (8) in [10, pg. 441], equation (97) in [16, pg. 28], equation (12) in [10, pg. 441], Equation (16) in [10, pg. 444]. Here, one unit of time equals $1/\mu$ seconds because IPP represent offered load, not arrival rate.

dynamic channels *offered* by cell x to the dynamic channels is taken to be a random variable representing the number of dynamic channels occupied in a $IPP(x)/M/\infty$ queue.[17] Using the approach in [14], the distribution of an $IPP(x)/M/\infty$ queue can be found, for $x \in IR(A_1)$.

B. Discounting/Aggregating Load Offered by Cells in BC to Dynamic Channels:

The total load offered to the dynamic channels by the cells in *BC* must be calculated. Let *bc* be the random variable which represents the number of dynamic channels which are locked in the cell A_1 by the cells in *BC*. Equivalently, let *bc* be the number of different dynamic channels in use in the $IR(A_1)$. The difficult part in determining the distribution of *bc* is accounting for the channel reuse among the cells in *BC*. This distribution can be obtained directly from the distributions of the random variables b_1, b_2, b_3, c_1, c_2, and c_3, (the distribution of $IPP(x)/M/\infty$ for $x \in BC$). A method which accounts for channel reuse among the cells in *BC* in its calculation of the distribution of *bc* is proposed for MP and SB below.

i. Maximum Packing (MP): As discussed in section II A, the condition for new call connection in MP is the RCC. The RCC and the CIC require that the total number of dynamic channels in use in each reuse cluster be less than or equal to $|D|$. In Fig. 8, $(a_1+b_i+c_j) \leq |D|$, where B_i and C_j are adjacent cells in the $IR(A_1)$.[18] There are six different reuse clusters in the $IR(A_1)$; $RC_1=\{A_1,B_1,C_1\}$, $RC_2=\{A_1,B_2,C_1\}$, $RC_3=\{A_1,B_2,C_2\}$, $RC_4=\{A_1,B_3,C_2\}$, $RC_5=\{A_1,B_3,C_3\}$, *and* $RC_1=\{A_1,B_1,C_3\}$.[19] Let rc_i be the random variable which represents the number of dynamic channels in use in RC_i in Fig. 8. Let L represent the number of dynamic channels which are locked in the cell A_1. Applying the RCC to each reuse cluster in Fig. 8 simultaneously, we obtain that $[L = \max(rc_i : i=1,..,6)]$ must be less than $|D|$ if a new call in cell A_1 is to be connected.[20] The a_1 random variable can be distributed out of (2) because it is present in all six of the reuse clusters. Here, $S_j = rc_j - a_1$ for $j = 1,..,6$. Thus, under MP, the distribution of *bc* can be found in terms of the distributions of the random variables b_1, b_2, b_3, c_1, c_2, and c_3, using (4-7).[21] The distributions of b_i, and c_i, $i=1,2,3$ were calculated in Section IV B. (6) uses the assumption of independence the among cells in *BC*.[22]

ii. Simple Channel Borrowing (SB): As discussed in section II B, it is reasonable to expect the SCCTP to hold when SB is used and the blocking probability is low. The SCCTP implies that the number of different dynamic channels in use in the cells B_i $i=1,2,3$, is equal to the max. of b_1, b_2, and b_3. Thus, given the SCCTP, $bc=\max(b_1,b_2,b_3)+\max(c_1,c_2,c_3)$. Assuming independence among cells in *BC*, (8-10) can be used to find the distribution of *bc* under SB.

$$L = \max(rc_j : j=1,..,6) = a_1 + bc \tag{2}$$

$$bc = \max(b_1+c_1,\ c_1+b_2,\ b_2+c_2,\ c_2+b_3,\ b_3+c_3,\ c_3+b_1) = \max(S_j : j \in \{1,..,6\} \tag{3}$$

$$P(bc=\alpha) = \sum_{\substack{s_j=0 \\ j\in\{1,..,6\}}}^{\alpha} P\left(S_j = s_j : j \in \{1,..,6\}\right) I_\alpha\left(s_j : j \in \{1,..,6\}\right) \tag{4}$$

$$I_\alpha\left(s_j : j \in \{1,..,6\}\right) = \begin{cases} 0, \text{ if } s_j \neq \alpha, \text{ for all } j \in \{1,..,6\} \\ 1, \text{ if } s_j = \alpha, \text{ for any } j \in \{1,..,6\} \end{cases} \tag{5}$$

---

[17] [14, Ch. 3]. The number of dynamic channels is infinite here so the offered (not carried) load from cell x can be found. The service rate is 1 because $E(t)$ represents offered load (Erlangs), not the arrival rate (calls per time unit).

[18] a_1 represents the offered load from cell A_1.

[19] The numbers of dynamic channels in use in the reuse clusters other than these six are unaffected by calls in the cell A_1. Thus, the RCC only considers channel use in these six reuse clusters.

[20] If $L=|D|$, then $L=|D|+1$ after the new call is connected in cell A_1, thus violating the RCC.

[21] This is a simple way of calculating the distribution of bc. More efficient and complex methods do exist.

[22] Independence assumption implies: $P(b_i=\beta_i,\ c_i=\delta_i : i \in \{1,2,3\}) = \prod_{i\in\{1,2,3\}} P(b_i=\beta_i)P(c_i=\delta_i)$

$$P\{S_j = s_j : j \in \{1..6\}\} = \sum_{\substack{\beta_i=0 \\ i\in\{1,..,3\}}}^{\Gamma_i} \sum_{\substack{\delta_k=0 \\ k\in\{1,..,3\}}}^{\Gamma_k} P(b_i = \beta_i) P(c_k = \delta_k) I_{MP}\left(\beta_i, \delta_k, s_j : \begin{matrix}(i,k) \in \{1.3\}\,\& \\ j \in \{1..6\}\end{matrix}\right) \tag{6}$$

$$I_{MP}(\beta_i, \delta_k, s_j : (i,k) \in \{1,2,3\}, j \in \{1..6\}) = \begin{cases} 1\text{: if} \begin{bmatrix} (\beta_1 + \delta_1) = s_1, \& (\beta_2 + \delta_1) = s_2, \& \\ (\beta_2 + \delta_2) = s_3, \& (\beta_3 + \delta_2) = s_4, \& \\ (\beta_3 + \delta_3) = s_5, \& (\beta_1 + \delta_3) = s_6 \end{bmatrix} \\ 0\text{: else} \end{cases} \tag{7}$$

$$P(bc = \alpha) = \sum_{\rho=0}^{\alpha} P(\max(b_1, b_2, b_3) = \rho) P(\max(c_1, c_2, c_3) = \alpha - \rho) \tag{8}$$

$$P(\max(b_1, b_2, b_3) = \rho) = \sum_{\substack{\beta_i=0 \\ i\in\{1,2,3\}}}^{\rho} P(b_i = \beta_i) I_{SB}(\beta_i, \rho : i \in \{1, 2, 3\}) \tag{9}$$

$$I_{SB}(\beta_i, \rho : i \in \{1, 2, 3\}) = \begin{cases} 1, \text{ if } \beta_i = \rho\text{: for any } i \in \{1, 2, 3\} \\ 0, \text{ if } \beta_i < \rho\text{: for all } i \in \{1, 2, 3\} \end{cases} \tag{10}$$

D. Find the Total Offered Load Rate from the BC Cells to the Dynamic Channels:

The assumption that the random variable *bc* was produced by a Poisson arrival stream is justified in section II.[23] Under this assumption, we can find the Poisson offered load parameter, N_L, directly. The Poisson (N_L) stream represents the load offered by the *BC* cells to the dynamic channels since the expected number of channels in use in a M/M/∞ queue with offered load (λ/μ), is (λ/μ), $N_L = E[bc] = \sum_{i=0}^{\infty} i * P[bc = i]$. The value of N_L is greater than each of the average IPP arrival rates from the individual cells in *BC*. However, because of the discounting of the IPP streams from the cells in *BC*, N_L is less than the sum of the average IPP arrival rates from the cells in *BC*.

E. The Blocking Probability of the Dynamic Channel Group:

This discounted aggregate stream (Poisson(N_L) and IPP(A_1)) is offered to the *set* of dynamic channels through the use of a *GI+M/M/|D|(|D|)*, GI=IPP(A_1), $M=N_L$. When all $|D|$ dynamic channels are locked in the cell A_1, the overflow stream from the fixed channels in the cell A_1 is blocked from the system. The Poisson (N_L) and IPP(A_1) streams see different blocking probabilities in the *GI+M/M/|D|(|D|)* queue. $q_{|D|}$ is the steady state probability that an arrival from the IPP stream will find all $|D|$ dynamic channels in use. Using the equations in [11], $q_{|D|}$ can be calculated using a piecewise Markov Process.

The final goal of this model is to obtain P_B, the probability that an arrival to the cell A_1 is blocked from the system. The Erlang-B formula, $B(F(A_1), a(A_1))$, in (4), equals the fraction of arrivals to the cell A_1 which are offered to the dynamic channel group (sections III and IVA). The $IPP(A_1)$ approximates the stream of arrivals to the cell A_1 which are offered to the dynamic channels. Thus, the probability that a new call in the cell A_1 is blocked from the system, P_B, is equal to $P_B = B(F(A_1), a(A_1)) \cdot q_{|D|}$.

V. Results

The models' results match well with those of a simulations in [8].[24] Due to space limitations, the plots are not included in this paper. In the simulation, the first dynamic

[23] This assumption can be eliminated. See section VI.

[24] The overall blocking probability of the GI+M stream was used in this comparison. The journal version of this paper will include a comparison which uses the IPP stream's individual blocking probability.

channel available was used to serve the calls which overflowed from the fixed channels. The model provides a better match with the simulation under light loading.

VI. Future Work

Different channel reassignment schemes could be incorporated into this approach by modifying the process of aggregating and discounting the load offered by each of the cells in *BC* (section IV C). One could also match the distribution of *bc*, with a generalized phase distribution instead of a Poisson distribution [13].[25] A generalized phase distribution would, by definition, provide a more accurate match. However, such an approach would increase the model's complexity. A generalized phase distribution would require the solution of a *discrete time Markov chain (DTMC)* to find the interarrival distribution of the composite stream.[26]

Because the RCC (SCCTP) is used by the MP (SB) model presented, only the channel usage in the cells in the $IR(A_1)$ is used by the model.[27] This approach can be extended to consider the effect of cells which are multiple cells (or IRs) away from the cell A_1. In single cell buffering, this can be done by first calculating the blocking probability in the cells which are x cells from the A_1.[28] Then, by iteratively using the blocking probability in these cells, one can discount the load offered by these cells to the dynamic channels in the cells which are x-1 cells away from the cell A_1. Using this approach, one may obtain more accurate results.

VII. Conclusion

In this paper, we presented a flexible performance analysis approach for HCA cellular voice networks. This approach is useful for the wireless network designer because it can analytically calculate the blocking probability in an HCA network which is using either SB or MP where the fixed to dynamic channel ratio and the offered load varies among the cells. The models presented are also valid in the presence of uneven loading among the cells. This performance analysis approach can be used to optimize the network by helping the system designer decide: 1) which simulations to run, 2) how many fixed channels allocate in each cell, 3) where to place cells, and 4) what size cells should be. This model is also interesting from a mathematical perspective because of its innovative use of the IPP.

VIII. References

[1] K. Araki, " Fundamental Problems of Nation-Wide Mobile Radio Telephone Systems." Review of the Electrical Communication Laboratory (Japan), 16(5-6):357-373, 5-6/1968.

[2] D. Cox and D. O. Reudink, "Effects of Some Nonuniform Spatial Demand Profiles on Mobile Radio System Performance," IEEE Trans. Vehic. Tech., Vol. 21, pp. 62-67, May 72.

[3] J. S. Engel and M. M. Peritsky, Statistically-optimum Dynamic Server Assignments in Systems with Interfering Servers," IEEE Trans. Vehic. Tech., 22(4), pp. 203-209, Nov. 73.

[4] D. Evritt and D. Manfield. "Performance Analysis of Cellular Mobile Communication Systems with Dynamic Channel Assignment," IEEE Journal on Selected Areas in Communications, vol. 7, no. 8, pp. 1172-1179, October 1989.

[5] S. Jordan, "Resource Allocation in Wireless Networks," Journal of High Speed Networks, Vol. 5, No. 1, 1996, pp. 23-34.

---

[25] One could quantify the amount of error introduced by the Poisson approximation of the *bc* stream by comparing the moments of the *bc* with those produced by the Poisson (N_1) stream.

[26] One would need to find the interarrival distribution of the aggregation of a the generalized phase distribution matched to *bc* and the $IPP(A_1)$ stream. Also, finding the blocking probability of the $IPP(A_1)$ stream in isolation would require additional consideration.

[27] We only block calls from the cells in *BC* when the IR(A_1) is full. For any cell x, $x \in BC$, an arrival to cell x is also blocked from the system if the $IR(x)$ is also full. This approximation is also supported by the observation that many of the dynamic channels locked in the cell A_1 are also locked in cell x, $x \in BC$.

[28] "x cells from the cell A_1," refers to cells which are $(S)(x)$ units from the cell A_1, (S = cell diameter).

[6] S. Jordan and A Khan, " A Performance Bound on Dynamic Channel Allocation in Cellular Systems: Unequal Load," preprint.
[7] S. Jordan and E. Schwabe, "Worst-Case Performance of Cellular Channel Assignment Policies", to appear in ACM Journal on Wireless Networks:
[8] T. J. Kahwa & N. D. Georganas, "A hybrid channel assignment scheme in large scale cellular structured mobile comm. systems," IEEE Transactions on Com., V-26, pp. 432, 1973.
[9] L. Kosten, Über Sperrungswahrsheinlichkeiten bei Stafffelschaltungen," Electro Nachrichten Technik, pp. 5-12, January 1937.
[10] Anatol Kuczura, "The Interrupted Poisson Process as an Overflow Process," Bell System Technical Journal, Vol. 52, pp. 437-448, 1973.
[11] Anatol Kuczura, "Loss Systems with Mixed Renewal and Poisson Inputs," Operations Research, vol. 21, pp. 787-795, 1973.
[12] Gabriel Montengro, "Time Dependent Analysis of Mobile Communication Traffic in a Ring Shaped Service Area with Nonuniform Vehicle Distribution," IEEE Transactions on Vehicular Technology, vol. 41, pp. 243-254, August 1992.
[13] J. Riordan, "Appendix: Derivation of Overflow traffic," Bell Sys. Tech. J., pg. 507, 6/56.
[14] L. Takàcs, "Introduction to the Theories of Queues," New York: Oxford University Press, Chapters 3 and 4, 1962.
[15] Roger I. Wilkinson, "Theories for Toll Engineering in the USA," The Bell System Technical Journal, pg. 421, March 1956.
[16] Ronald Wolff, "Stochastic Modeling and the Theory of Queues," Englewood Cliffs, New Jersey: Prentice Hall, 1989.
[17] Wuyi Yue, "Analytical Methods to Calculate the Performance of a Cellular Mobile Radio Communication System with HCA," IEEE Transactions on Vehic. Tech., Vol. 40, May 1991.

APPENDIX A: SIMPLE CHANNEL BORROWING (SB) EXAMPLE

Assume 9 channels are partitioned and initially assigned as described in Table 1. Suppose that a single new call is placed, in order, in each of the following cells: A_1, B_1, C_1, and C_1. The resulting channel usage after these four new calls are connected is shown in table 1 under the SB scheme and under a non-SB scheme.[29] When the first new call is placed in cell A_1 none of the channels in $F(A_1) \cup N(A)$ are available in A_1. If SB is employed, channel 8 is borrowed from N(C), such that $\min(Avail(x): x \in BC) = 1$ after the channel is borrowed. If SB is not employed (table 2), channel 6 is borrowed from N(B), such that $\min(Avail(x): x \in BC) = 0$ after the channel is borrowed. When SB is used, the SCCTP is found to be true. However, in the non-SB case, the SCCTP is violated by the use of channel 6 in both C_2 & A_1.

Cell	*Fixed*	*Nominal*	*Initial*	*After Calls with SB*	*After Calls without SB*
A_1	1	2,3	1,2,3	1,2,3,***8***	1,2,3,***6***
B_1	4	5,6	4,5	4,5,***6***	4,5,***8***
C_2	7	8,9	7	7,8,9	7,9,***6***

Table 1

It is possible that the SCCTP property will not hold when SB is employed. This is more likely to occur under heavy loading. For example, if $|D|$ calls are placed both in cell B_1 and in cell C_2, then the SCCTP will not hold even if SB is used. However, if $(|D|/2)$ calls are placed both in cell B_1 and in cell C_2, then the SCCTP will hold if SB is used.

[29] Channels allocated for the new calls are indicated by *italics*, and the borrowed channels are shown in **bold.**

The Multicovering Radii of Codes Extended Abstract

Andrew Klapper*
University of Kentucky
Dept. of Computer Science
763H Anderson Hall
Lexington, KY, 40506-0046
klapper@cs.engr.uky.edu.

Abstract

The covering radius of a code is the least r such that the set of balls of radius r around codewords covers the entire ambient space. We introduce a generalization of the notion of covering radius. The *m-covering radius* of a code is the least radius such that the set of balls of the radius covers all m-tuples of elements in the ambient space. We investigate basic properties of m-covering radii. We investigate whether codes exist with given m-covering radii (they don't always). We derive bounds on the size of the smallest code with a given m-covering radius, based on generalizations of the sphere bound. More precise calculations are given for Hamming codes.

1 Introduction – Basic Concepts

The *covering radius* of a block code C is the smallest radius such that the set of balls of that radius covers the ambient space. This concept has been the subject of a large body of research in the past couple of decades. See [2] for a comprehensive survey and thorough bibliography on the subject. In this paper we investigate simultaneous coverings of m-tuples of vectors, rather than single vectors. The *m-covering radius* of C is the smallest radius such that every m-tuple of vectors in the ambient space is contained in at least one ball of that radius around some codeword. More precisely, if m is a positive integer, then the m-covering radius $t_m(C)$ of a code C of length n is the smallest integer t such that for every collection S of m vectors of length n, there is a code word v in C whose distance from every u in S is at most t. Thus for $m = 1$ we have the ordinary covering radius of C. At times we will also refer to $t_m(C)$ as a *multicovering radius* of C.

The basic questions concerning multicovering radii concern bounds on certain code parameters given others. For given length, m, and t, what is the smallest length n code with $t_m = t$? What if we constrain the minimum distance or require that C be maximum or maximal? As with the ordinary covering radius, precise general answers to these

*Project sponsored by the National Science Foundation under grant number NCR-9400762.

questions are unlikely. Thus we concentrate on finding upper and lower bounds and in exploring relationships among the answers for different sets of parameters.

The notion of multicovering radius arose from investigations concerning the cryptanalysis of stream ciphers. Recall that a stream cipher is determined by a binary sequence B that is XORed with the message. In various cryptanalytic attacks an algorithm is used to determine an efficient generator for B given a small number of bits of B [4, 6]. We recently investigated whether there exist efficient generators of sequences that resist all possible attacks of this general type [5]. (The answer is yes, but the results give no help in finding an efficient construction for such a family of sequences.) More generally, we considered attacks in which the attacker only hopes to correctly generate a substantial fraction of the bits of B. Using properties of the covering radius of Reed-Muller codes, we showed that there exists an efficiently generated family of sequences that is secure against all such attacks in the sense that, for each such attack, there are infinitely many sequences in the family that resist the attack. A better result would be that the family resists all such attacks in the sense that, for each such attack, all but finitely many of the sequences resists the attack. Extending the techniques used seems to depend on finding good bounds for the multi-covering radii of efficiently generated codes. The idea is to show that for any set S of m sequences, there is an efficiently generated sequence B that is far from all the sequences in S. This is equivalent to saying that the complement of B is close to every sequence in S, hence is a statement that an m-covering radius is small. Beyond these questions, multi-covering radii are interesting in their own right as natural generalizations of the covering radius.

2 Notation and Terminology

As much as possible, we have attempted to make our notation consistent with that used by Cohen, Litsyn, Lobstein and Mattson in their recent survey paper [2]. Thus we have: $\mathbf{N}$ = the set of natural numbers; $\mathbf{F} = GF(2) = \{0, 1\}$.

For $x \in \mathbf{F}^n$: $\mathrm{supp}(x) = \{i : x_i \neq 0\}$. The complement of x is denoted x' ($x'_i = 1 - x_i$). The Hamming weight of x is $\mathrm{wt}(x) = |x| = |\mathrm{supp}(x)|$. The weight of a set S is the maximum of the weights of its elements,

$$\mathrm{wt}(S) = \max\{\mathrm{wt}(x) : x \in S\}.$$

Also,

$$B_t(x) = \{y \in \mathbf{F}^n : d(x, y) \leq t\}$$

and

$$V(n, t) = |B_n(t)|.$$

For $x, y \in \mathbf{F}^n$, $d(x, y) = \mathrm{wt}(x + y)$, the distance between x and y. This is the number of coordinate locations in which x and y differ. If S is a set, then

$$d(x, S) = \min\{d(x, y) : y \in S\}.$$

The covering radius of x for S is

$$cov(x, S) = \max\{d(x, y) : y \in S\},$$

the size of the smallest ball centered at x that contains all of S. If C is a code, the covering radius of C for S is

$$cov(C, S) = \min\{cov(c, S) : c \in C\},$$

the size of the smallest ball centered at some x in C that contains all of S. Thus the covering radius of C is

$$t_m(C) = \max\{cov(C,S) : |S| = m\}.$$

The concatenation of x and y is denoted $(u|v)$.

A set C is said to (m,t)-*cover* (or simply cover if there is no ambiguity) a set V if for all $x^1, \cdots, x^m \in V$, there is a $c \in C$ such that $\forall i = 1, \cdots, m : d(x^i, c) \leq t$. Such a C is called an (m,t)-covering of V.

$t_m(C) = m$-covering radius of C.

$t_m[n,k] =$ smallest m-covering radius among all $[n,k]$ codes.

$t_m(n,k) =$ smallest m-covering radius among all (n,k) codes.

$k_m[n,t] =$ smallest dimension of a linear code of length n and m-covering radius t.

$K_m(n,t) =$ smallest cardinality of a code of length n and m-covering radius t.

2.1 The Translate Leader

If C is a code, $c \in C$, and $S = (v_1, \cdots, v_m)$ is a set of vectors, then

$$cov(C,S) = cov(C, S+c),$$

where $S + c = \{x + c : x \in S\}$. A *translate* of C is a set of m-tuples, $S +_m C = \{S + c : c \in C\}$, and a *translate leader* is an m-tuple $T \in S +_m C$ such that $\mathrm{wt}(T)$ is minimal. The m-covering radius of C is the weight of the maximal weight translate leader.

3 Basic Relationships

Certain basic relations hold as we vary the parameters for covering radii. The proofs are straightforward.

Proposition 3.1

1. *If* $C_1 \subseteq C_2$, *then* $t_m(C_1) \geq t_m(C_2)$.
2. *For any code* C *and* $m \in \mathbf{N}^*$, $t_m(C) \leq t_{m+1}(C)$.
3. *For any* $n, m, k \in \mathbf{N}^*$, $t_m(n,k) \leq t_{m+1}(n,k)$ *and* $t_m[n,k] \leq t_{m+1}[n,k]$.
4. *For any* $n, m, k \in \mathbf{N}^*$, $t_m(n,k) \geq t_m(n,k+1)$ *and* $t_m[n,k] \geq t_m[n,k+1]$.
5. *For any* $n, m, t \in \mathbf{N}^*$, $K_m(n,t) \leq K_{m+1}(n,t)$ *and* $k_m[n,t] \leq k_{m+1}[n,t]$.
6. *For any* $n, m, t \in \mathbf{N}^*$, $K_m(n,t) \geq K_m(n,t+1)$ *and* $k_m[n,t] \geq k_m[n,t+1]$.

We next consider the relationship between the multicovering radii of two codes and various codes that can be built from them. For $i = 1, 2$, let C_i be an $[n_i, k_i, d_i]$ code.

3.1 Cartesian Product

Let

$$C = C_1 \times C_2 = \{(x|y) : x \in C_1, y \in C_2\}.$$

This code has type $[n_1 + n_2, k_1 + k_2, \min\{d_1, d_2\}]$. If S is a set of m vectors in $\mathbf{F}^{n_1+n_2}$, then the sets of projections on the first and second components are within $t_m(C_1)$ and $t_m(C_2)$ of some code words in C_1 and C_2, respectively. Thus

$$t_m(C) \leq t_m(C_1) + t_m(C_2). \tag{1}$$

When $m = 1$ this inequality becomes an equality. However, for $m \geq 2$, it may be strict. For example, if $C_1 = C_2 = \{(0,0),(0,1)\}$, then $t_2(C_i) = 2$, while $t_2(C_1 \times C_2) = 3$.

On the other hand, suppose S_i is a set of m_i vectors of length n_i whose distance to C_i equals a_i $(i = 1, 2)$. Let $S = \{(x|y) : x \in S_1, y \in S_2\}$. Then the distance from S to C is $a + b$. Thus

$$t_{m_1 m_2}(C) \geq t_{m_1}(C_1) + t_{m_2}(C_2).$$

The example in the preceeding paragraph shows that this inequality can be strict. The same inequality holds for the catenation of two linear codes since it is a subcode of their Cartesian product.

3.2 $(u, u + v)$ Construction

Suppose $n_1 = n_2$ and $C_2 \subseteq C_1$. The code C defined by

$$C = \{(u, u + v) : u \in C_1, v \in C_2\}$$

is a $[2n_1, k_1 + k_2, \min\{2d_1, d_2\}]$ code (see [7][Ch. 2, Sec. 9]).

Lemma 3.2 *The code C satisfies*

$$t_{m^2}(C) \geq 2t_m(C_1).$$

Proof: Let T be a translate leader of C_1 of maximum weight. Let $S = \{(a, b) : a, b \in T\}$, so S has cardinality m^2. We claim that S is a translate leader of its translate. If not, then there exist $u \in C_1$ and $v \in C_2$ such that $\mathrm{wt}(S + (u, u + v)) < \mathrm{wt}(S)$. Equivalently,

$$\max\{\mathrm{wt}(a + u) + \mathrm{wt}(b + u + v) : a, b \in T\} < max\{\mathrm{wt}(a) + \mathrm{wt}(b)\}.$$

The right hand side of this inequality is just $2\mathrm{wt}(T)$, while the left hand side is the sum of the weights of two other elements of the translate of T. This contradicts the minimality of $\mathrm{wt}(T)$, proving the claim. This also shows that $\mathrm{wt}(S) = 2\mathrm{wt}(T)$, proving the lemma. □

3.3 Repetition

For any integer $r \in \mathbf{N}^*$, the *r-fold repetition* of C_1 is the code

$$C = \{(c|c|\cdots|c) : c \in C_1\},$$

where the code word c is repeated r times. This is a $[rn_1, k_1, rd_1]$ code with $t_m(C) = rt_{rm}(C_1)$.

3.4 Lengthening a Code

A linear code, C, can be lengthened by adjoining a column h to its $k \times n$ generator matrix G, giving rise to the generator matrix $G; h$. The new code is an $[n+1, k]$ code C'. It's codewords consist of vectors of the form $(c, c \cdot b)$, where c is a codeword of C and b is a fixed vector, a new parity check. For any m, the m-covering radius of C' is either $t_m(C)$ or $t_m(C)+1$. We wish to determine when the m-covering radius increases.

Lemma 3.3 *Suppose the code C is lengthened to C' by the addition of a parity check b. Then*

$$t_m(C') = 1 + t_m(C)$$

if and only if there is a translate leader $S = \{v_1, \cdots, v_m\}$ of C of weight $t_m(C)$ and a vector $(e_1, \cdots, e_m) \in \mathbf{F}^m$ such that whenever $c \in C$ and $S + c$ is a translate leader, we have $e_i \neq c \cdot b$ for some i such that $\mathrm{wt}(v_i + c)$ is maximal.

Proof: Suppose $S = \{v_1, \cdots, v_m\}$ is an m-tuple of vectors in $\mathbf{F}^n$. S gives rise to 2^m m-tuples in $\mathbf{F}^{n+1}$, each of the form $S_e = \{(v_1, e_1), \cdots, (v_m, e_m)\}$ for some vector $e = (e_1, \cdots, e_m) \in \mathbf{F}^m$. The weight of S_e can be at most one greater than the weight of S. Thus the m-covering radius increases when the parity check is added if and only if the weight of every translate leader of some maximal weight translate increases. For given S and e, the translate of S_e consists of m-tuples of the form $\{(v_1 + c, e_1 + c \cdot b), \cdots, (v_m + c, e_m + c \cdot b)\}$ where $c \in C$. The weight of this m-tuple is greater than that of $S + c$ exactly when $e_i \neq c \cdot b$ for some i such that $\mathrm{wt}(v_i + c)$ is maximal. □

Corollary 3.4 *Appending a zero parity or an overall parity check to a code increases the m-covering radius by 1.*

Proof: In the first case, take $e_1 = \cdots = e_m = 1$.

In the second case, for any set S, let S' be the odd parity vectors in S and let S'' be the even parity vectors. The maximum weight vectors in S must either all be in S' or all in S''. Suppose S is a maximum weight translate leader and we are in the latter case. We let $e_i = 0$ if $v_i \in S'$, and $e_i = 1$ if $v_i \in S''$. Let $T = S + c$ have the same weight as S. The maximum weight elements of T are in T''. If c has even weight, then $T'' = S'' + c$. For such a vector, $e_i = 1 \neq c \cdot b$. Similarly, if c has odd weight, then $T'' = S' + c$. For such a vector, $e_i = 0 \neq c \cdot b$. If the maximum weight vectors of S are in S', we simply reverse the definition of the e_i. □

4 Lower Bounds

As we have seen, the m-covering radius of a fixed code C, $t_m[n,k]$, $t_m(n,K)$, $k_m[n,t]$, and $K_m(n,t)$ are nondecreasing functions of m. Thus a lower bound for, say, $K_m(n,t)$ implies a bound for $K_{m+1}(n,t)$. Our first bound shows that for $m \geq 2$, the situation for m-covering radii is quite different from that for ordinary covering radii.

Proposition 4.1 *If $m \geq 2$, then for any C,*

$$t_m(C) \geq \lceil n/2 \rceil .$$

Proof: We may assume $m = 2$. Let t be the m-covering radius of C. Let x be any vector of length n, and let $S = \{x, x'\}$. Then for any c,

$$d(x,c) + d(c,x') \geq d(x,x') = n.$$

It follows that one of $d(x,c)$ and $d(x',c)$ is at least $n/2$ □

Thus $K_m(n,t)$ is undefined if $t < n/2$. When this is the case we say $K_m(n,t) = \infty$. There are other circumstances when $K_m(n,t)$ is undefined. For example,

$$K_{2^n}(n, n-1) = \infty.$$

More generally, $K_m(n,t) = \infty$ if $m > V(n,t)$, since in this case no ball of radius t covers any set of m distinct vectors. We would like to know whether $K_m(n,t)$ is finite for given n, m, t. This is the case if and only if $t_m(\mathbf{F}^n) \leq t$, since the m-covering radius of $\mathbf{F}^n$ lower bounds the m-covering radii of all other codes of dimension n. Thus we want to determine $t_m(\mathbf{F}^n)$. We show that it in fact differs from $n/2$ by a constant depending on m but not depending on n.

Theorem 4.2 *For every m there is a constant r_m such that, for every n,*

$$t_m(\mathbf{F}^n) \leq \left\lceil \frac{n + r_m}{2} \right\rceil.$$

Proof Sketch: We prove this by induction on n, using the idea behind equation (1). For a given m-tuple of vectors in $\mathbf{F}^n$, $U = \{u_1, \cdots, u_m\}$, we need to decompose $\mathbf{F}^n$ and U with it so that the resulting pair of m-tuples are covered by balls of small radius. The base case needs to be chosen large enough so that such a decomposition is always possible in the induction step. If n is large enough, some pair of columns contains at most three binary pairs, so this pair of columns can be covered with radius one. □

It is apparent from the proof of the preceeding theorem that the better we can estimate the smallest possible value of L, the better we can bound r_m. The crude estimate in the proof gives $r_m \leq 2^m + 1$. For example, any set of vectors containing a column with at most one 1 or at most one 0 must have a pair of indices with the property in the proof. Thus we can improve the bound to $r_m \leq 2^m - 2m - 1$. The following theorem gives a lower bound.

Theorem 4.3 *For every m and n satisfying $m \leq 2^n$,*

$$t_m(\mathbf{F}^n) \geq \left\lceil \frac{n + \lfloor \log_n(m) \rfloor - 1}{2} \right\rceil.$$

Proof Sketch Let

$$k = \lfloor \log(m) \rfloor,$$

so that

$$2^k \leq m < 2^{k+1}.$$

For any $v \in \mathbf{F}^k$, let

$$w_v = \begin{cases} v|0^{n-k} & \text{if wt}(v) \text{ is even,} \\ v|1^{n-k} & \text{if wt}(v) \text{ is odd.} \end{cases}$$

A set of m vectors in $\mathbf{F}^n$ that contains every w_v satisfies the bound. □

This inequality is an equality for $2 \leq m \leq 5$. However, Iiro Honkala [3] has given an example that shows the inequality can be strict.

4.1 The Sphere Bound

We generalize the classical sphere bound, first noted by van Tilborg [9]

Theorem 4.4 *For any (n, K) code C,*

$$K \geq \frac{\binom{2^n}{m}}{\binom{V(n,t_m(C))}{m}}.$$

Thus for any n, t, m,

$$K_m(n,t) \geq \frac{\binom{2^n}{m}}{\binom{V(n,t)}{m}}.$$

Proof: We prove this bound by counting unordered sets of distinct m-tuples of vectors. Each such m-tuple must occur in a neighborhood of radius $t_m(C)$ around at least one code word. The number of m-tuples is 2^n choose m, while the number of m-tuples in a neighborhood of radius $t_m(C)$ is $V(n, t_m(C))$ choose m. □

Corollary 4.5 *If*

$$\binom{2^n}{m} > 2^n \binom{V(n,t)}{m}$$

then $K_m(n,t) = \infty$.

One can also count ordered m-tuples (distinct or not), or not necessarily distinct unordered m-tuples. If we count distinct ordered m-tuples, we get the same bound. In the other two cases we get weaker bounds.

5 Large Values

In this section we consider bounds in various cases of large values of the parameters. When $t = n$, every codeword covers every vector, so a code of size 1 will m-cover $\mathbf{F}^n$ for every m.

When $t = n - 1$, any code of size $m + 1$ will m-cover $\mathbf{F}^n$. To prove this we first need a lemma.

Lemma 5.1 *Let x be a natural number, $x > 0$, and let m and k be natural numbers. Define*

$$f_{k,m}(x) = \sum_{i=0}^{k} (-1)^i \binom{k}{i} (x-i)^m.$$

Then $f_{k,m}(x) = 0$ if and only if $k \geq m + 1$.

Theorem 5.2 *For all natural numbers n and m, we have*

$$K_m(n, n-1) = m + 1.$$

Proof: Let C be any code. Each neighborhood of radius $n-1$ around a codeword c omits precisely one vector, the complement of c. Thus the intersection of the neighborhoods of radius $n-1$ around any j codewords omits precisely j vectors. If $|C| = k$, it follows from the inclusion/exclusion principle that the number of m-tuples of vectors covered by the neighborhoods of radius $n-1$ around the code words of C is precisely

$$2^{mn} - f_{k,m}(2^n).$$

The theorem follows immediately from Lemma 5.1. □

Corollary 5.3 *For all natural numbers n, $t \leq n-2$, and m, we have*

$$K_m(n,t) \geq m+1.$$

6 Repetition and Hamming Codes

As a trivial example, the m-covering radius of the repetition code $C = \{0^n, 1^n\}$ is n, since the smallest ball around a codeword that covers the 2 element set $\{0^n, 1^n\}$ has radius n.

It is well known that the ordinary covering radius of a Hamming code is 1, i.e. minimal [7]. Here we prove that the m-covering radius of a Hamming code is nearly minimal. We denote the rth Hamming code by H_r. This is a $[2^r-1, 2^r-r-1, 3]$ code, with the $r \times (2^r-1)$ matrix whose columns are all nonzero vectors of length r as parity check matrix. It follows that for some ordering of the coordinates of H_r,

$$H_r = \{(v+u, u, a) : v \in H_{r-1} \text{ and } a = \mathrm{wt}(u) \bmod 2\}.$$

When $r = 2$, H_r is just the repetition code of length 3, so $t_m(H_2) = 3$ if $m \geq 2$.

Proposition 6.1 *For any $m \geq 2$, there is a constant c_m such that for any $r \geq 2$ we have*

$$2^{r-1} \leq t_m(H_r) \leq 2^{r-1} + c_m r.$$

Proof Sketch: The lower bound is from Proposition 4.1. The upper bound is proved by induction, using the recursive definition of Hamming codes and Theorem 4.2. □

For $m = 4$ or 5, we can take $r_m = 1$. For $m = 3$, we have $\lceil r_m/2 \rceil = 0$. In this case we can improve the induction to show that

$$2^{r-1} \leq t_m(H_r) \leq 2^{r-1} + 1.$$

For $m = 2$ we have a precise answer.

Proposition 6.2 *For any $r \geq 3$,*

$$t_2(H_r) = 2^{r-1}.$$

Proof Sketch: The proof is by induction on r. We prove the stronger statement that any pair of vectors in $\mathbf{F}^{2^r-1}$ is covered by at least two distinct codewords of H_r within radius 2^{r-1}. This can be done by considering different cases of pairs of vectors in H_r, varying the parity of and distance between the vectors. □

7 Conclusions and Open Problems

We have introduced a natural generalization of the covering radius of a code. We have described many of the basic properties of multicovering radii. In many cases this was a matter of generalizing earlier results on covering radii to this new setting. The subject of covering radii is large and one can attempt to duplicate all the work that has been done on this subject in this new setting. In particular, we would like better bounds on various fundamental quantities: the size of the smallest code with specified length and m-covering radius; the minimum m-covering radius for a code a certain size and length.

We would like to see precise values for these quantities for small length codes. This is, to a large extent, a matter of working out the m-covering radii for many of the standard codes. We have begun this sort of analysis by considering Hamming codes, but our analysis even in this case is not complete. It is apparent from what has been done here that the situation for multicovering radii is far more complicated than for ordinary covering radii.

Finally, this work was motivated by work in the cryptanalysis of stream ciphers. It became apparent that to prove an existence theorem for stream ciphers secure against a very general type of attack we need a class of efficiently generated codes that have small multicovering radii. Hamming codes satisfy the second requirement but, unfortunately, not the first. Reed-Muller codes of bounded degree satisfy the first requirement, but, as yet, we do not know whether they satisfy the second. Moreover, known bounds on their ordinary covering radii were sufficient to acquire some useful cryptologic results. Thus we consider the determination of sharp bounds on the multicovering radii of Reed-Muller codes to be a major open problem.

References

[1] G.D. Cohen, M.G. Karpovsky, H.F. Mattson, Jr., and J.R. Schatz, "Covering radius – survey and recent results," *IEEE Trans. Info. Theory* vol. IT-31, pp. 328-343, 1985.

[2] G.D. Cohen, S.N. Litsyn, A.C. Lobstein, and H.F. Mattson, Jr., "Covering Radius 1985-1994," Technical Report 94 D 025, Dept. Informatique, Ecole Nationale Supérieure des Télécommunications, 1994.

[3] I. Honkala, private communication.

[4] A. Klapper, "The Vulnerability of Geometric Sequences Based on Fields of Odd Characteristic," *Journal of Cryptology* vol. 7 pp. 33-51, 1994.

[5] A. Klapper, "On the Existence of Secure Feedback Registers," in *Advances in Cryptology: Proc. Eurocrypt '96, Lecture Notes in Computer Science Vol. 388*, Springer-Verlag: Berlin, pp. , 1996.

[6] A. Klapper and M. Goresky, "Feedback Shift Registers, Combiners with Memory, and 2-Adic Span," to appear, *Journal of Cryptology.*

[7] F.J. MacWilliams and N.J.A. Sloane, *The Theory of Error Correcting Codes,* North-Holland: Amsterdam, 1977.

[8] J.L. Massey, "Shift register sequences and BCH decoding," *IEEE Trans. Info. Theory* vol. IT-15, pp. 122-127, 1969.

[9] H.C.A. van Tilborg, *Uniformly Packed Codes.* Eindhoven, The Netherlands: Eindhoven Tech. Univ., 1976.

[10] G.J.M. van Wee, "Improved sphere bounds on the covering radius of codes," *IEEE Trans. Info. Theory* vol. 34, pp. 237-245, 1988.

Convolutional Codes, r-Cyclic Codes, and Linear Systems.

Eric V. York*
Department of Mathematics
University of Notre Dame
Notre Dame, IN 46556
e-mail: Eric.V.York@nd.edu

October 4, 1996

Abstract

In this paper we explore the relationship between systems theory and coding theory. In particular, we examine autonomous behaviors and their connection to r-cyclic (quasi-cyclic) codes. The main result of this short note is a structure theorem for r-cyclic codes in terms of their state space representations.

1 Introduction

It is well known that error correcting codes can be viewed as linear systems over a finite field. The first authors to take advantage of the dynamical structure of certain classes of error correcting codes were J. Massey and M. Sain in [12, 13]. In these papers the authors view convolutional codes and cyclic codes as linear systems described by state space equations of the form

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ y_t &= Cx_t + Du_t \end{aligned}$$

and relate the properties of the codes to properties of the matrices A, B, C, D that define the codes. In recent years there has been a resurgence of interest in the connections between error correcting codes and linear systems theory (see e.g. [2, 3, 5, 6, 9, 15, 17, 18]).

One important class of block codes are the so called quasi-cyclic codes which we will refer to as r-cyclic codes. This class of codes generalizes the class of cyclic codes and it has been shown that there exist r-cyclic codes that meet the Gilbert-Varshamov bound [7]. In [16], r-cyclic codes were shown to be convolutionally encodable, hence one

*The author is supported in part by a fellowship from the Center for Applied Mathematics of the University of Notre Dame

could use sequential or Viterbi decoding with this class of codes. One could also use this relationship to construct convolutional codes as presented in [8, 11].

In this short note we will generalize the results of Massey and Sain in [12] concerning cyclic codes to the class of r-cyclic codes. We show how this view gives us new insight into the structure of r-cyclic codes as well as their relation to convolutional codes at the state space level.

In Section 2 we review the relevant definitions and basic properties of r-cyclic and convolutional codes. Section 3 generalizes the results in [12] to r-cyclic codes. Using this generalization we derive some results concerning the structure of r-cyclic codes. In Section 4 we show how certain choices of matrices in the state space form of a r-cyclic code lead to MDS r-cyclic codes. Finally in Section 5 we explore the connection between r-cyclic codes and convolutional codes on the state space level.

2 Definitions and Basic Properties

For the remainder of this note let $\mathbb{F} = \mathbb{F}_q$ be the Galois field with q elements.

Definition 2.1 Let $N, K \in \mathbb{N}$ with $K < N$, and let $\mathcal{C}$ be a submodule of rank K contained in $\mathbb{F}^N[z]$. Then we say that $\mathcal{C}$ is a rate $\frac{K}{N}$ convolutional code.

The above definition of a convolutional code is slightly different from the standard definition as presented in [4] in that we only consider the code as being generated by all finite (polynomial) information sequences as opposed to infinite (Laurent series) input. The subtle differences between these definitions are explored in greater detail in [15].

Let $n, k \in \mathbb{N}$ with $k < n$, and recall that a block code is defined as a k dimensional (linear) subspace of $\mathbb{F}^n$.

Definition 2.2 Let $\mathcal{C}$ be an $\frac{k}{n}$ block code defined over $\mathbb{F}$. Let $c = (c_1, c_2, \ldots c_n) \in \mathcal{C}$. We say that $\mathcal{C}$ is r *-cyclic* if its code words are invariant under the permutation $c_i \to c_{i+r \bmod n}$ where r is the smallest such integer. We say that $\mathcal{C}$ is an $[n, k]_r$code. If the minimum distance of $\mathcal{C}$ is d then we say $\mathcal{C}$ is an $[n, k, d]_r$ code.

Since r is the smallest such integer, we must have $n = rm$ for some $m \in \mathbb{N}$. *r-cyclic codes* are also known as *quasi cyclic* codes (see e.g. [10] or [14]).

Throughout this paper we will make use of the vector space isomorphism $\mathbb{F}^n \cong \frac{\mathbb{F}[z]}{<z^n-1>}$. In this way a code $\mathcal{C}$ can be seen as a subset of $\frac{\mathbb{F}[z]}{<z^n-1>}$, where each $c \in \mathcal{C}$ is identified with the polynomial $\sum_{i=0}^{n-1} c_i z^i$ For ease of notation we make the following:

Definition 2.3 Let $n, r \in \mathbb{N}$ then define

$$S_{n,r} := \frac{\mathbb{F}[z^r]}{< z^n - 1 >}.$$

With this we obtain the following:

Lemma 2.4 *Let $\mathcal{C}$ be a $[n, k]_r$ code. Then $\mathcal{C}$ is an $S_{n,r}$ sub-module of $S_{n,1}$. Conversely, every $S_{n,r}$ sub-module of $S_{n,1}$ corresponds to a $[n, k]_{\tilde{r}}$ code for some $\tilde{r} \mid n$.*

As a consequence, we obtain the well known structure theorem for cyclic codes:

Corollary 2.5 *If $\mathcal{C}$ is cyclic, then it is an ideal in $S_{n,1}$.*

Convolutional codes can be given a very similar module structure as well. To see this let $\mathcal{C}$ be the convolutional code defined by the generator

$$G(z) = \begin{pmatrix} g_{11} & g_{12} & \cdots & g_{1K} \\ g_{21} & g_{22} & \cdots & g_{2K} \\ \vdots & \vdots & \ddots & \vdots \\ g_{N1} & g_{N2} & \cdots & g_{NK} \end{pmatrix}.$$

Lemma 2.6 *Every convolutional code $\mathcal{C}$ can be viewed as a $S_{0,N}$ sub-module of $S_{0,1}$ generated by:*

$$\{\sum_{i=1}^{N} z^{i-1} g_{ij}(z^N) \mid 1 \leq j \leq k\}.$$

Hence, convolutional codes and r-cyclic codes have a very similar module structure. We can gain some insight as to why they are similar using linear systems theory.

3 Codes as Linear Systems

From a systems theory point of view, codes can be viewed as the behavior of a discrete-time linear system defined over a finite field. Recall:

Definition 3.1 The system

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ y_t &= Cx_t + Du_t \end{aligned} \tag{3.1}$$

defines a *Finite State Machine* (FSM) , where $t \in \mathbb{N}$ is the time variable, x_t is an s-dimensional vector called the *state vector*, u_t is a i-dimensional vector called the *input vector* and y_t is a r-dimensional vector called the *output vector*. A, B, C, and D are matrices over $\mathbb{F}_q$ with dimensions $s \times s, s \times i, r \times s$, and $r \times i$ respectively.

It is well known that state space systems of the form (3.1) correspond to convolutional codes (see e.g. [12] or [15] for generalized representations of convolutional codes). Our primary interest in this section will be generalizing the results of [12] concerning cyclic codes, to $r-$cyclic codes.

By setting $u_t = 0, \forall t \in \mathbb{N}$ system (3.1) reduces to

$$\begin{aligned} x_{t+1} &= Ax_t \\ y_t &= Cx_t \end{aligned} \tag{3.2}$$

which can be expressed in terms of the initial state vector as

$$x_{t+1} = A^t x_0 \tag{3.3}$$

$$y_t = CA^t x_0. \tag{3.4}$$

We denote the output of (3.4) up to time m relative to the initial condition x_0 by

$$Y_m(x_0) = [y_0^t, y_1^t, \cdots y_{m-1}^t]. \tag{3.5}$$

We say that the above system is *zero input periodic* if their exists a $m \in \mathbb{N}$ such that $y_{t+m} = y_t$ for all $t \in \mathbb{N}$ and all $x_0 \in \mathbb{F}_q^n$. The smallest such m is called the period.

Theorem 3.2 *Given the FSM (3.2) with period m, then $\{Y_m(x_0) \mid x_0 \in \mathbb{F}_q^n\}$ is isomorphic to a r-cyclic subspace $V \in \mathbb{F}_q^{rm}$ with $\dim(V) \leq s$. Furthermore, if the pair $[A, C]$is observable with observablity index $< m$, then A must be non-singular and the system given by (3.2) can be used as an encoder for a rate $\frac{s}{rm}$ r-cyclic code, where our information vectors are taken as the initial conditions.*

In particular, the r-cyclic code has the $rm \times s$ generator matrix G given by

$$G = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{m-1} \end{pmatrix} \tag{3.6}$$

Conversely we have the following:

Theorem 3.3 *For every $[rm, k]_r$ code $\mathcal{C}$ there corresponds a zero-input finite-state machine of the form (3.2). Furthermore, there exists a choice of the pair $[A, C]$ with a non-singular $k \times k$ matrix A and a $r \times k$ matrix C such that $[A, C]$ is observable.*

The representation given by $[A, C]$ is in general not unique. In fact for a $T \in Gl_k$ we have the natural equivalence given by

$$[A, C] \sim [T^{-1}AT, CT]. \tag{3.7}$$

Equation (3.7) provides the key tool for understanding the structure of r-cyclic codes. In particular, it follows that any quasi-cyclic code can be defined by an A, C pair where A is given by

$$A = \begin{pmatrix} A_1 & & & \\ & A_2 & & \\ & & \ddots & \\ & & & A_\nu \end{pmatrix} \tag{3.8}$$

where each A_i is a matrix in Frobenius (rational canonical) form. That is

$$A_i = \begin{pmatrix} 0 & 0 & \cdots & \cdots & 0 & a_0 \\ 1 & 0 & & & \vdots & a_1 \\ 0 & 1 & \ddots & & \vdots & a_2 \\ \vdots & 0 & \ddots & \ddots & \vdots & \vdots \\ \vdots & \vdots & \ddots & \ddots & 0 & \vdots \\ 0 & 0 & \cdots & 0 & 1 & a_{d_i} \end{pmatrix}, \tag{3.9}$$

and ν is the number of irreducible cyclic components of A. From this we derive the following:

Lemma 3.4 *Let $\mathcal{C}$ be an r-cyclic code defined by the pair of matrices A, C. Then $\mathcal{C}$ has a cyclic basis if the Frobenius form of A has only one block. In particular, if $e_1 = (1, 0, \ldots, 0)^t$ is the $k \times 1$ elementary column vector and G is given by (3.6), then $\mathcal{C}$ is generated by the vector Ge_1 and its first $k-1$, r-cyclic shifts.*

Such a code is called an irreducible r-cyclic code. Using Lemma 3.4, we obtain the following basic structure theorem for r-cyclic codes (compare to [1, Theorem 4]:

Theorem 3.5 *Let $\mathcal{C}$ be an r-cyclic code defined by the pair of matrices A, C. Where A is given by (3.8) and C is partitioned accordingly as $C = (C_1, C_2, \ldots, C_\nu)$. Then $\mathcal{C}$ is isomorphic to the direct sum of irreducible codes defined by the pairs A_i, C_i for $i = 1, 2, \ldots, \nu$.*

Define the z-transform of the sequence $\mathbf{x} = (x_0, x_1, x_2, \cdots\cdots)$ by

$$z\mathbf{x} = (0, x_0, x_1, x_2, \cdots\cdots). \tag{3.10}$$

That is, the z-transform is just the right shift operator on the sequence space $(\mathbb{F}^r)^{\mathbb{N}}$. By applying (3.10) to the systems defined by (3.2) or (3.1) one can extract the generators for the module (or ideal in the case of cyclic codes) of the underlying code. If the matrix A is in Frobenius form, the determination of the generators becomes much simplier.

4 MDS r-Cyclic Codes

Let $\alpha \in \mathbb{F}^*$ and $k, r \in \mathbb{N}$. Consider the following two matrices.

$$A = \begin{pmatrix} \alpha^r & 0 & \cdots & 0 \\ 0 & \alpha^{2r} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \alpha^{kr} \end{pmatrix} \tag{4.1}$$

$$C = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ \alpha & \alpha^2 & \cdots & \alpha^k \\ \alpha^2 & \alpha^4 & \cdots & \alpha^2 k \\ \vdots & \vdots & & \vdots \\ \alpha^{r-1} & \alpha^{2(r-1)} & \cdots & \alpha^{k(r-1)} \end{pmatrix} \tag{4.2}$$

Lemma 4.1 *Choose $\alpha \in \mathbb{F}^*$ and let $\operatorname{ord}\alpha = rm$ in the multiplicative group $\mathbb{F}^*$. Choose $k \in \mathbb{Z}_+$ such that $k < rm$. Let A and C be defined as in (4.1) and (4.2) . Then the matrix*

$$\begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{m-1} \end{pmatrix} \tag{4.3}$$

has the property that every subset of k rows are linearly independent.

Theorem 4.2 *Choose $\alpha \in \mathbb{F}^*$ and let $\operatorname{ord}\alpha = rm$ in the multiplicative group of $\mathbb{F}^*$. For $k \in \mathbb{Z}_+$ such that $k < rm$ the code defined by (4.1) and (4.2) is a $[rm, k, d]_r$ code with $d = rm - k + 1$.*

In [15] the authors use a similar techniques to construct convolutional codes.

5 Relations to Convolutional Codes

In [16] the authors show that to any convolutional code, one can associate a whole family of r-cyclic codes. In fact, when one considers the module structure of each code, this result is expected. In this section we will show how this correspondence goes on the state space level for rate $\frac{1}{N}$ memory s convolutional code. Let $\mathcal{C}$ be the convolutional code given by

$$\begin{aligned} x_{t+1} &= Ax_t + Bu_t \\ y_t &= Cx_t + Du_t \end{aligned} \tag{5.1}$$

where

$$A = \begin{pmatrix} 0 & 0 & \cdots & 0 \\ 1 & \ddots & \ddots & \vdots \\ 0 & \ddots & \ddots & 0 \\ \vdots & \ddots & 1 & 0 \end{pmatrix} \in \mathbb{F}^{s\times s}, \quad B = \begin{pmatrix} 1 \\ 0 \\ \vdots \\ 0 \end{pmatrix} \in \mathbb{F}^{s\times 1} \tag{5.2}$$

and $C \in \mathbb{F}^{N\times s}$ is chosen so that the system is observable, (i.e. the code is non-catastrophic). Then we have the following:

Theorem 5.1 *The shortest r-cyclic code associated to the convolutional code $\mathcal{C}$ is defined by the pair*

$$\widetilde{A} = \left(\begin{array}{c|ccc} 0 & 0 & \cdots & 0 \quad 1 \\ \hline B & & A & \end{array}\right), \widetilde{C} = (C, D)\,.$$

6 Conclusions

In this paper we showed how r-cyclic codes can be viewed as autonomous linear systems. We also demonstrated that the structure of the underlying code was related to the matrices A and C that define the code. In the last section we showed how convolutional and r-cyclic codes are related as finite state machines. The results in this short note are worked out in greater generality and detail in [19].

References

[1] J. Conan and G. Séguin. Structural properties and enumeration of quasi cyclic codes. *Applicable Algebra in Engineering, Communication and Computing*, AAECC:25–39, 1993.

[2] E. Fornasini and M.E. Valcher. Algebraic aspects of 2D convolutional codes. *IEEE Trans. Inform. Theory*, IT-40(4):1068–1082, 1994.

[3] E. Fornasini and M.E. Valcher. Observability and extendability of finite support nD behaviors. In *Proc. of the 34th IEEE Conference on Decision and Control*, pages 3277–3282, New Orleans, Louisiana, 1995.

[4] G. D. Forney. Convolutional codes I: Algebraic structure. *IEEE Trans. Inform. Theory*, IT-16(5):720–738, 1970.

[5] G. D. Forney, B. Marcus, N. T. Sindhushayana, and M. Trott. A multilingual dictionary: System theory, coding theory, symbolic dynamics and automata theory. In *AMS Proceedings of Symposia in Applied Mathematics*. American Mathematical Society, 1995. To appear.

[6] G. D. Forney and M. D. Trott. The dynamics of group codes: State space, trellis diagrams and canonical encoders. *IEEE Trans. Inform. Theory*, IT-39:1491–1513, 1993.

[7] T. Kasami. A Gilbert-Varshamov bound for quasi-cyclic codes of rate 1/2. *IEEE Trans. Inform. Theory*, IT-20:679, 1974.

[8] Y. Levy and D.J. Costello Jr. An algebraic approach to constructing convolutional codes from quasi-cyclic codes. *DIMACS Series in Discrete Mathematics and Theoretical Computer Science*, 14:189–198, 1993.

[9] H. A. Loeliger, G. D. Forney, T. Mittelholzer, and M. D. Trott. Minimality and observability of group systems. *Linear Algebra Appl.*, 205/206:937–963, 1994.

[10] F. J. MacWilliams and N. J.A. Sloane. *The Theory of Error-Correcting Codes*. North Holland, Amsterdam, 1977.

[11] J. L. Massey, D.J. Costello, and J. Justesen. Polynomial weights and code constructions. *IEEE Trans. Inform. Theory*, IT-19(1):101–110, 1973.

[12] J. L. Massey and M. K. Sain. Codes, automata, and continuous systems: Explicit interconnections. *IEEE Trans. Automat. Contr.*, AC-12(6):644–650, 1967.

[13] J. L. Massey and M. K. Sain. Inverse of linear sequential circuits. *IEEE Trans. on Computers*, C-17(4):330–337, 1968.

[14] W. Peterson and E. J. Weldon. *Error Correcting Codes.* M.I.T. Press, Cambridge, MA, 2nd edition, 1972.

[15] J. Rosenthal, J.M. Schumacher and E.V. York. On behaviors and convolutional codes. To appear in *IEEE Trans. Inform. Theory*,1996.

[16] G. Solomon and H.C.A. van Tilborg. A connection between block and convolutional codes. *SIAM J. APPL. MATH.*, 37:358–369, 1979.

[17] M.E. Valcher and E. Fornasini. On 2D finite support convolutional codes: an algebraic approach. *Multidim. Sys. and Sign. Proc.*, 5:231–243, 1994.

[18] E.V. York, J. Rosenthal, and J. M. Schumacher. On the relationship between algebraic systems theory and coding theory: Representations of codes. In *Proc. of the 34th IEEE Conference on Decision and Control*, pages 3271–3276, New Orleans, Louisiana, 1995.

[19] E.V. York. Algebraic Description and Construction of Error Correcting Codes, a Systems Theory Point of View. Ph.D. Thesis, University of Notre Dame, in preparation.

Analysis of Convolutional Encoders via Generalized Sylvester Matrices and State Space Realizations *

Brian M. Allen
Department of Mathematics
University of Notre Dame
Notre Dame, Indiana 46556-5683, USA
e-mail: ballen1@artin.helios.nd.edu

Joachim Rosenthal
Department of Mathematics
University of Notre Dame
Notre Dame, Indiana 46556-5683, USA
e-mail: Joachim.Rosenthal@nd.edu

October 4, 1996

Abstract

Using results from systems theory, we develop two methods for determining if a full rank $k \times n$ polynomial matrix $G(s)$ has coprime full size minors and even to determine the greatest common divisor (GCD) of the full size minors. This has the immediate application of determining whether a convolutional encoder is observable or not. We will also be able to calculate a syndrome former and polynomial inverse (when one exists) for $G(s)$. The first method uses generalized Sylvester matrices to compute the GCD of two polynomial matrices. The second method involves finding a realization of the system given by $\ker G(s)$ and then finding the rank of the controllability matrix of the state space representation. Both of these methods seem to have better complexities than existing methods.

1 Introduction.

There are few reliable constructions for finding convolutional codes and encoders, hence brute force computer searches have been the main tools for finding them. It is naturally desirable in these searches to be able to determine very quickly which encoders are *noncatastrophic*. Remember, an encoder is called noncatastrophic if there does not exist input with arbitrarily large or even infinite weight that becomes encoded as a finite weight codeword. It is well known that convolutional encoders *i.e.* $k \times n$ polynomial matrices with full rank, are noncatastrophic if and only if the GCD of the full size minors are of the form s^α for some nonnegative integer α. Often, it is even desirable for the full size minors to be coprime. In this case, the code and encoder are called *observable* .

Since the brute force calculation of the GCD of the full size minors involves a lot of polynomial operations, it becomes infeasible to incorporate into a massive computer search. Indeed, let us consider a 10×20 polynomial matrix with a degree 10 term in each row. One would potentially have to calculate 184,756 10×10 determinants, each being a polynomial of degree up to 100, and then still have to calculate their GCD! Hence we have sought efficient methods for determining if an encoder is noncatastrophic and even observable, which can be effectively implemented in a computer search.

*This research was supported in part by NSF grant DMS-94-00965 and the Arthur J. Schmitt Foundation.

A recent attempt in this area was made by Koorapaty *et. al.* [5]. Their time-domain approach is effective in determining if an encoder is observable and also for determining polynomial inverses and syndrome formers. Its shortcoming lies in that it cannot determine the GCD of the full size minors and hence is unable to distinguish between catastrophic encoders and noncatastrophic, nonobservable encoders. Another drawback is that the approach is based on graph theoretic ideas rather than the algebraic properties of the encoder.

Our first method incorporates these ideas with the solid algebraic ideas of Kung *et. al.* [7, 8, 1] for finding the greatest common divisor of two polynomial matrices, which relies heavily on Forney's work with minimal bases[3].

Our second method uses recent results of Schumacher and Rosenthal [11] for obtaining realizations for systems given in terms of a behavior. Its efficiency comes from the fact that the realization is obtained with little or no calculation, *i.e.* by inspection.

2 Minimal Bases.

Let $G(s)$ be a $k \times n$ polynomial matrix over an arbitrary field $\mathbb{F}$. Assume further that $G(s)$ has full row rank. Denote the rows of $G(s)$ by $\mathbf{g_i}(s)$, with $1 \leq i \leq k$.

Definition 2.1 [3]

(a) The i^{th} *row degree* of $G(s)$ is defined as the maximum degree of the polynomial components of $\mathbf{g_i}(s)$ and is denoted by ν_i.

(b) The *order*, or total degree, of $G(s)$ is denoted d, and is defined as $d := \sum_{i=1}^{k} \nu_i$.

(c) If $V \subseteq \mathbb{F}^n(s)$ is a k-dimensional vector space, then a *minimal basis* of V is a $k \times n$ polynomial matrix $G(s)$ that is a basis for V and has minimal order among polynomial bases of V.

We now state (most of) Forney's main theorem on minimal bases.

Theorem 2.2 [3] *The following are equivalent:*

(1) G is a minimal basis for V

(2) a) The full size minors of G are coprime, and b) the high order coefficient matrix of G, G_∞, i.e. the matrix obtained by taking the coefficients of the highest degree appearing in each row, has full rank.

(3) If $\mathbf{y} = \mathbf{x}G$ is a polynomial n-tuple, then a) $\mathbf{x}$ must be a polynomial k-tuple, and b) deg $\mathbf{y} = max_{1 \leq i \leq k}(deg\ x_i + \nu_i)$

(4) The indices ν_i are such that for all $j \geq 0$, $\dim_{\mathbb{F}} V_j = \sum_{\nu_i \leq j}(j - \nu_i)$, where V_j is the vector space over $\mathbb{F}$ of polynomial n-tuples in V of degree less than j. □

These are the driving ideas behind our approach. Indeed, let G be a (not necessarily minimal) basis of the k-dimensional vector space $V \subseteq \mathbb{F}^n(s)$, and consider the map

$$\begin{aligned} \phi : \mathbb{F}^n(s) &\longrightarrow \mathbb{F}^k(s) \\ u &\longmapsto Gu \end{aligned} \tag{2.1}$$

Let $\ker \phi = V^{\perp}$ and let H be a minimal basis for $V^{\perp}$ over $\mathbb{F}(s)$. Let the column degrees (or indices) of H be μ_i, $1 \leq i \leq n-k$, and the order of H be $d^{\perp}$. By part (4) of Theorem 2.2, $\dim_{\mathbb{F}} V_j^{\perp} = \sum_{\mu_i \leq j}(j - \mu_i)$. For j larger than the greatest μ_i we have $\dim_{\mathbb{F}} V_j^{\perp} = (n-k)j - d^{\perp}$. The key now is that if G is a minimal basis we must have $d = d^{\perp}$ [3]. Hence G is a minimal basis if and only if

$$\dim_{\mathbb{F}} V_j^{\perp} = (n-k)j - d \tag{2.2}$$

Kung *et. al.* [7, 8, 1] have developed an algorithm which calculates $\dim_{\mathbb{F}} V_j^{\perp}$ and applies it, in a slightly different context, as we shall see, to determine the veracity of equation (2.2).

3 A First Approach.

In this section we review the concept of divisors of a matrix as well as some very important results of Kung *et. al.* [7, 8, 1]. We then use these results to develop an algorithm for determining if our matrix $G(s)$ is observable, and even to compute the GCD of the full size minors. Further, we will be able to compute a syndrome former for $G(s)$ and, when G is observable, to compute a polynomial inverse.

3.1 Matrix GCDs via Generalized Sylvester Matrices

Given a polynomial matrix $A(s)$ with k rows, a (left) *divisor* of $A(s)$ is a $k \times k$ polynomial matrix $R(s)$ such that there exists a polynomial matrix A_1 satisfying $A = RA_1$. Given another matrix $B(s)$ with k rows we may speak of *common* left divisors and *greatest* common left divisors (GCLDs), defined in the obvious way. We say A and B are (left) coprime if their GCLD is a unimodular matrix.

What is of interest to us is that given our matrix $G(s)$, we partition G into $G = [A \mid B]$, where A is $k \times (n-k)$ and B is $k \times k$. In this situation the divisors of G are precisely the common divisors of A and B. By our assumption that G has full row rank we have that all GCLDs of A and B must be nonsingular and differ only by unimodular (right) factors [4, Lemma 6.3-4].

Proposition 3.1 *The full size minors of $G(s)$ are coprime if and only if A and B are left coprime. In fact, the GCD of the full size minors of $G(s)$ equals the determinant of $R(s)$, the GCLD of A and B.*

Proof: The divisors of G are exactly the common divisors of A and B. Further, a divisor of G must also be a common divisor of all $k \times k$ submatrices of G. Hence, the determinant of any divisor of G must also be a common factor of all full size minors. Clearly, the determinant of a GCLD will be the greatest of such. □

It is in this setting that Kung *et. al.* [7, 8, 1] have developed their algorithm. (Although Kung considered greatest common right divisors instead of left divisors, clearly the theory holds *mutatis mutandis.*) To be more precise, they use equation (2.2) to determine whether A and B are left coprime. The algorithm, which is a natural generalization for the Sylvester resultant method for determining the GCD of two polynomials, goes on to explicitly determine a GCLD. Our slightly different angle is to start with our $k \times n$ full rank matrix $G(s)$, and then to divide G into matrices A and B as before. Then the algorithm will, by the proposition, tell us if $G(s)$ is an observable encoder and even will compute explicitly the GCD of the full size minors of $G(s)$. The algorithm can then be extended to compute a syndrome former for G and a polynomial inverse when one exists.

3.2 An Algorithm by Matrix GCDs

Assume we have a full rank $k \times n$ polynomial matrix $G(s)$ divided into submatrices A and B as before. Write $G(s) = \sum_{i=0}^{\ell} G_{\ell-i} s^i$. For j=1,2,..., we have

$$S_j = \begin{bmatrix} G_0 & 0 & \cdots & 0 \\ G_1 & G_0 & \ddots & \vdots \\ \vdots & G_1 & \ddots & 0 \\ G_\ell & \vdots & \ddots & G_0 \\ 0 & G_\ell & & G_1 \\ \vdots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & G_\ell \end{bmatrix} \quad \text{(j column blocks)} \tag{3.1}$$

whose dimensions are $k(\ell + j) \times jn$. These are the generalized Sylvester matrices. Let the rank of S_j be r_j.

We wish to calculate r_j for increasing values of j. There are many ways for computing the rank of a scalar matrix, but we suggest the following way which takes advantage of the shifted structure of the matrix and also enables us to calculate the GCLD of A and B.

For each j, we 'column reduce' S_j using only the following operations: i) scalar multiplication of columns, ii) adding a multiple of one column to another within each column block, and iii) adding a multiple of one column to another column in a column block further to the right. The end result of the column reduction is a (possibly column permuted) column echelon form for the matrix, which we denote E_j. Note that, due to the structure of S_j, 'column reducing' S_{j+1} is not a separate task. In fact, once we 'column reduce' S_j to E_j, we need only to 'column reduce' n more columns to compute E_{j+1} *i.e.*, all but the last n columns of E_{j+1} will be the same as E_j with k extra rows of zeroes added onto the bottom.

We also wish to keep track of the column operations we perform on S_j to arrive at E_j. In particular, we want the matrix M_j, such that $S_j M_j = E_j$. Again note that calculating M_{j+1} only involves computing an additional n columns to be attached to the right of M_j. Also, it should be clear by our allowed operations that M_j must be an upper block diagonal matrix.

Our algorithm proceeds as follows, starting with $j = 0$ and $r_0 = 0$:

STEP 1. Compute E_{j+1}, M_{j+1}, and r_{j+1}. (r_{j+1} will be the number of nonzero columns of E_{j+1}.) If $r_{j+1} - r_j = k$ then proceed to step 2. Otherwise, let $j := j+1$ and repeat step 1.

STEP 2. At this point, compute $m := r_j - kj$. G will be observable if and only if m is the highest degree of the full size minors [8]. However, the highest degree is not easily computed unless we put the additional assumption that G_∞ is full rank, *i.e.* G is row proper. In that case, the highest degree of a full size minor will be exactly d, the order of G.

Without this additional assumption, and indeed for many other purposes, it is beneficial to carry the algorithm even further.

If the leading terms in each of the k nonzero columns in the last block of E_{j+1} are in rows of different residue classes mod k then stop. Otherwise calculate E_{j+2} and M_{j+2}, and so on, until this condition holds.

Assume we have continued the process until the condition was finally satisfied for E_t.

Theorem 3.2 [1] *The last k nonzero columns of E_t give the GCLD, $R(s)$, of A and B.* □

Remark 3.3 The matrix R is read off from these columns by taking the first k rows on the bottom to be the degree zero terms; the second k rows from the bottom give the coefficients of degree one terms, and so on.

As we have already seen, the determinant of R is the GCD of the full size minors of G.

We can now also calculate syndrome formers and polynomial inverses.

Theorem 3.4 *The columns in M_t corresponding to the zero columns in the last column block of E_t give a syndrome former for G.*

Proof: This follows clearly from what the matrix equation $S_t M_t = E_t$ represents. Note that this matrix is read off from M_t similar as to how R is read off from E_t. □

Now if R is unimodular, then G is observable and we should expect to find a polynomial inverse for G. Indeed, since R is unimodular, it is column equivalent to the identity. If we permit these column operations on E_t and keep track of these operations in M_t, we get the matrices $\bar{E}_t$ and $\bar{M}_t$.

Theorem 3.5 *The k columns of $\bar{M}_t$ corresponding to the columns of $\bar{E}_t$ which give the identity, will, when read off in the usual fashion, give a polynomial inverse for G.*

Proof: Clear. □

Remark 3.6 If we are only interested in whether or not $G(s)$ is observable or in calculating the GCD of the full size minors then we may remove each zero column as it occurs and not bring it down to the next S_j. This will save on the memory required and will marginally increase the speed of the algorithm. It will, however, excessively complicate the matter of finding syndrome formers and polynomial inverses.

4 A Second Approach.

In this section we attack the same problem from a different angle. In particular, we review some results on realization in systems theory to obtain, without calculation, a state space representation of a system. From this system we can efficiently determine the rank of the so-called controllability matrix to determine if the system is controllable, and hence if our original matrix is observable.

4.1 Realization.

We now review some results of realization theory. For a more extensive account of this subject we refer the reader to [11, 6, 12, 13, 14]. Let $\mathbb{F}[[t]]$ be the space of sequences over the field $\mathbb{F}$. Hence, $\mathbb{F}^n[[t]]$ is the space of sequences over $\mathbb{F}^n$. Now, given a $k \times n$ polynomial matrix $G(s)$, the *behavior* associated with $G(s)$ is defined by:

$$\mathcal{B}(G) \;=\; \{w \in \mathbb{F}^n[[t]] \mid G(\sigma)w = 0\} \tag{4.1}$$

where σ is the left shift operator.

Remark 4.1 It is clear that elementary row operations on $G(s)$ do not change $\mathcal{B}(G)$. Further, if $G(s)$ and $\hat{G}(s)$ are full row rank polynomial matrices then $\mathcal{B}(G) = \mathcal{B}(\hat{G})$ if and only if there exists a unimodular matrix $U(s)$ such that $\hat{G}(s) = U(s)G(s)$.

The behavior associated with a triple of matrices (M, N, P) (with sizes $\ell \times (\ell + n - k), \ell \times (\ell + n - k)$, and $n \times (\ell + n - k)$ respectively) is given by:

$$\mathcal{B}(M, N, P) = \{w \in \mathbb{F}^n[[t]] \mid \exists\, z \in \mathbb{F}^{\ell+n-k} \text{ s.t. } M\sigma(z) = Nz,\ w = Pz\} \tag{4.2}$$

The triple (M, N, P) is a *realization* of $G(s)$ if $\mathcal{B}(M, N, P) = \mathcal{B}(G)$.

The key here is that when (M, N, P) is minimal (as a pencil representation), *i.e.* when $\ell + n - k$ is minimal among all such representations of a behavior, we can, without any calculation, obtain from the matrices (M, N, P) a standard state space representation of the system given as the behavior of $G(s)$. In particular, it is possible, under conditions we will describe later, to arrive at the following, where A, B, C, and D are the usual matrices in a state space representation [11].

$$\begin{bmatrix} sN - M \\ P \end{bmatrix} = \begin{bmatrix} sI - A & B \\ C & D \\ 0 & I \end{bmatrix} \tag{4.3}$$

We now present the main result we will need, proceeded by some notation. For a more thorough account of the ideas involved, please refer to [11].

Partition $G(s)$ into $G(s) = [\,E(s) \mid F(s)\,]$, where $E(s)$ is $k \times k$ and $F(s)$ is $k \times n - k$. Assume $G(s)$ is row proper with $\nu_1 \geq \cdots \geq \nu_k \geq 1$. For $i, j = 1, \ldots, k$ let

$$e_{i,j}(s) = \sum_{\alpha=0}^{\nu_i} e_{i,j}^{\alpha} s^{\alpha} \tag{4.4}$$

denote the polynomial entries of E(s). Also let

$$\mathbf{f_i}(s) = \sum_{\alpha=0}^{\nu_i} \mathbf{f_i}^{\alpha} s^{\alpha} \tag{4.5}$$

denote the i^{th} row of $F(s)$. Define for $i = 1, \ldots, k$ matrices of sizes $\nu_i \times \nu_i$, $\nu_i \times n - k$ and $1 \times \nu_i$ respectively:

$$A_{i,i} := \begin{bmatrix} 0 & \cdots & \cdots & \cdots & -e_{i,i}^{0} \\ 1 & 0 & & & -e_{i,i}^{1} \\ 0 & 1 & \ddots & & \vdots \\ \vdots & & \ddots & 0 & \vdots \\ 0 & \cdots & 0 & 1 & -e_{i,i}^{\nu_i - 1} \end{bmatrix},\ B_i := \begin{bmatrix} \mathbf{f_i}^{0} \\ \mathbf{f_i}^{1} \\ \vdots \\ \mathbf{f_i}^{\nu_i - 1} \end{bmatrix},\ C_i := [0, \ldots, -1]. \tag{4.6}$$

For $i, j = 1, \ldots, k$, $i \neq j$ define matrices of size $\nu_i \times \nu_j$:

$$A_{i,j} := \begin{bmatrix} 0 & \cdots & 0 & -e_{i,j}^{0} \\ \vdots & & \vdots & -e_{i,j}^{1} \\ \vdots & & \vdots & \vdots \\ 0 & \cdots & 0 & -e_{i,j}^{\nu_i - 1} \end{bmatrix} \tag{4.7}$$

Remark 4.2 These matrices are called the (observable) companion forms of a state space system by Wolovich [15], and have a very familiar form. In particular, the matrices $A_{i,i}$ are just the companion matrices for the polynomials $e_{i,i}$. The matrices $A_{i,j}$ are companion matrices without the off-diagonal of $1's$. To get these forms we require certain conditions which are given below.

Theorem 4.3 [11] *Given $G(s)$ as above, and also satisfying $G_\infty = [\, I_k \mid 0 \,]$ then*

$$x(t+1) = \begin{bmatrix} A_{1,1} & \cdots & A_{1,k} \\ \vdots & \ddots & \vdots \\ A_{k,1} & \cdots & A_{k,k} \end{bmatrix} x(t) + \begin{bmatrix} B_1 \\ \vdots \\ B_k \end{bmatrix} u(t), \tag{4.8}$$

$$y(t) = \begin{bmatrix} C_1 & & 0 \\ & \ddots & \\ 0 & & C_k \end{bmatrix} x(t) \tag{4.9}$$

represents a minimal state space realization of the system

$$E(\sigma)y(t) + F(\sigma)u(t) = 0. \tag{4.10}$$

□

As usual, we call the first matrix in (4.8) A, and the second matrix B. We see that A has size $d \times d$ (where d is, as before, the order of $G(s)$). Also the size of B is $d \times (n-k)$.

Remark 4.4 The very important idea here is that controllability of the state space representation is equivalent to the controllability of the behavioral system given by $G(s)$, which is equivalent to G being an observable encoder [10, 11, for example].

We now describe the obvious algorithm which arises from Theorem (4.3).

Given a $k \times n$ full rank polynomial matrix $G(s)$, partition G into matrices E and F as above. Next perform the necessary row and column operations on G to bring the matrix into the form required by the theorem. Typically, a computer search could be designed to search only for these types of matrices, so this step is usually not necessary.

Next, apply the theorem to find the matrices A and B. Now compute the controllability matrix $C(A, B)$:

$$C(A,B) = [\, B \; AB \; A^2B \; \ldots \; A^{d-1}B \,] \tag{4.11}$$

Finally, check the rank of this matrix. The rank of $G(s) = d$ if and only if then the pair (A,B) is controllable and hence the matrix $G(s)$ is observable.

Remark 4.5 This algorithm has by no means been streamlined for efficiency. Future work on this algorithm will provide details on how to implement this algorithm more efficiently and also on how this method can be extended to compute the GCD of the full size minors.

5 Examples.

We have provided some examples to help clarify the algorithms.

Example 5.1 Consider the rate $\frac{2}{4}$ encoder over the field $\mathbb{F} = GF(2)$

$$G(s) = \begin{bmatrix} s^2+1 & 1 & s+1 & 1 \\ s & s^2 & s & 1 \end{bmatrix} \tag{5.1}$$

We will evaluate this encoder using our first algorithm. We see that S_1 is given by

$$S_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.2}$$

Clearly $E_1 = S_1$ and hence M_1 is the identity matrix. We proceed to the end of the algorithm and present the matrices M_4 and E_4 which appear in the equation $S_4 M_4 = E_4$:

$$M_4 = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ & & & & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & 0 & 1 & 1 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ & & & & 0 & 0 & 1 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\ & & & & & & & & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ & & & & & & & & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ & & & & & & & & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 1 \\ & & & & & & & & 0 & 0 & 1 & 0 & 1 & 1 & 0 & 1 \\ & & & & & & & & & & & & 0 & 0 & 1 & 1 \\ & & & & & & & & & & & & 0 & 0 & 0 & 1 \\ & & & & & & & & & & & & 1 & 1 & 1 & 0 \\ & & & & & & & & & & & & 0 & 1 & 0 & 0 \end{bmatrix} \tag{5.3}$$

$$E_4 = \begin{bmatrix} 1 & 0 & 0 & 0 & & & & & & & & & & & & \\ 0 & 1 & 0 & 0 & & & & & & & & & & & & \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & & & & & & & & \\ 1 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & & & & & & & & \\ 1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & & & & \\ 0 & 0 & 0 & 1 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & & & & \\ & & & & 1 & 1 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ & & & & & & & & & & & & 1 & 0 & 0 & 0 \\ & & & & & & & & & & & & 0 & 1 & 0 & 0 \end{bmatrix} \tag{5.4}$$

From here we can see that $r_4 = 12$ and $r_3 = 10$, hence $r_4 - r_3 = 2 = k$ so we stop our algorithm. We see that $m := r_3 - k(3) = 10 - 6 = 4$. Since $G(s)$ is row proper we see that the highest degree of a full size minor is 4, hence $G(s)$ is observable. Also, we are already in a position to determine (explicitly) the GCLD as well as a syndrome former

and polynomial inverse. The last two nonzero columns of E_4 show that the GCLD is given by

$$R(s) = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \tag{5.5}$$

From this again we can see that $G(s)$ is observable. Further, no more reductions are needed to determine an inverse. From columns 13 and 14 of M_4 we see that an inverse is given by

$$G^{-1}(s) = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 1 \\ s & s+1 \end{bmatrix} \tag{5.6}$$

Also the last two columns of M_4 provide a syndrome former:

$$G^{\perp} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ s^2+s+1 & s \\ s^3+s^2 & s \end{bmatrix} \tag{5.7}$$

Example 5.2 We return to the encoder given in example 5.1. We will evaluate this same encoder using our realization algorithm. It is clear that we have $G_\infty = [I_2 \mid 0]$. It is easy to find our matrices A and B as well as the controllability matrix.

$$A = \begin{bmatrix} 0 & 1 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \qquad B = \begin{bmatrix} 1 & 1 \\ 1 & 0 \\ 0 & 1 \\ 1 & 0 \end{bmatrix} \tag{5.8}$$

$$C(A,B) = \begin{bmatrix} \mathbf{1} & \mathbf{1} & 0 & 0 & 0 & 0 & \mathbf{1} & \mathbf{1} \\ \mathbf{1} & 0 & \mathbf{1} & \mathbf{1} & 0 & 0 & 0 & 0 \\ 0 & \mathbf{1} & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathbf{1} & 0 & \mathbf{1} & \mathbf{1} & \mathbf{1} & \mathbf{1} & 0 & 0 \end{bmatrix} \tag{5.9}$$

We can see right away that $\operatorname{rank} C(A,B) = 4$ so our encoder is observable.

Example 5.3 Clearly the size of our example is too small to show the teeth of the realization algorithm, so let us return to our example of a 10×20 matrix with a degree 10 polynomial in each row. Our realization algorithm requires that we determine the rank of a 100×1000 scalar matrix. In practice, however it can be shown that no more than 109 columns need to be analyzed. While we don't recommend that you try this by hand, it is clearly vastly superior to computing 184,756 polynomial minors, some with degrees as high as 100 and then finding their GCD!

6 Conclusion.

In this paper we have used tools and techniques from systems theory to develop methods for attacking a basic question in convolutional coding theory, *i.e.* is a given encoder observable, non-catastrophic, or catastrophic? Our first method makes use of results on matrix GCDs and generalized Sylvester matrices. It partitions the matrix $G(s)$ into two matrices and computes the GCLD of these matrices. The determinant of which is the GCD of the full size minors of G. The second method uses results on obtaining realizations of behavioral systems. A state space representation is found by inspection and then we evaluate the controllability matrix of this representation to determine the observability of G. The efficiency of these methods follows from the lack of polynomial operations involved.

References

[1] R. R. Bitmead, S. Y. Kung, B. D. O. Anderson, and T. Kailath. Greatest common divisors via generalized Sylvester and Bézout matrices. *IEEE Trans. Automat. Control*, AC-23:1043–1047, 1978.

[2] G. D. Forney. Convolutional codes I: Algebraic structure. *IEEE Trans. Inform. Theory*, IT-16(5):720–738, 1970.

[3] G. D. Forney. Minimal bases of rational vector spaces, with applications to multivariable linear systems. *SIAM J. Control Optim.*, 13(3):493–520, 1975.

[4] T. Kailath. *Linear Systems.* Prentice-Hall, Englewood Cliffs, N.J., 1980.

[5] H. Koorapaty, A. Dholakia, D. Bitzer, and M. Vouk. Rate-k/n convolutional encoders: A time-domain analysis. Technical Report TR-95-04, North Carolina State University, 1995.

[6] M. Kuijper and J. M. Schumacher. Realization of autoregressive equations in pencil and descriptor form. *SIAM J. Control Optim.*, 28(5):1162–1189, 1990.

[7] S.-Y. Kung. *Multivariable and multidimensional systems: Analysis and design.* PhD thesis, Dep. Elec. Eng. Standford University, 1977.

[8] S.-Y. Kung, T. Kailath, and M. Morf. A generalized resultant matrix for polynomial matrices. In *Proc. IEEE Conf. on Decision and Control (Florida)*, pages 892–895, 1976.

[9] Ph. Piret. *Convolutional Codes, an Algebraic Approach.* MIT Press, Cambridge, MA, 1988.

[10] M. S. Ravi, J. Rosenthal, and J. M. Schumacher. Homogeneous behaviors. Report BS-R9607, CWI, Amsterdam, June 1996.

[11] J. Rosenthal and J. M. Schumacher. Realization by inspection. CWI Report BS-R9534, December 1995.

[12] J. C. Willems. Input–output and state–space representations of finite–dimensional linear time–invariant systems. *Linear Algebra Appl.*, 50:581–608, 1983.

[13] J. C. Willems. From time series to linear system. Part III: Approximate modelling. *Automatica*, 23:87–115, 1987.

[14] J. C. Willems. Paradigms and puzzles in the theory of dynamical systems. *IEEE Trans. Automat. Control*, AC-36(3):259–294, 1991.

[15] W. A. Wolovich. *Linear Multivariable Systems*, volume 11 of *Appl. Math. Sc.* Springer Verlag, New York, 1974.

Long-Term Dependent VBR Video Model Using ON-OFF Sources

Qiwei Xiao and Edward J. Coyle
School of Electrical and Computer Engineering
Purdue University, West Lafayette, IN 47907-1285
{qiwei, coyle}@ecn.purdue.edu

Abstract

Empirical studies have indicated that variable-bit-rate(VBR) video sources exhibit self-similar behavior. Since this behavior can not be produced by traditional Poisson-based traffic models, new models have been proposed. They achieve strict-self-similarity by relying on fractional ARIMA processes, nonlinear chaotic maps, or aggregation of ON-OFF sources with sub-exponential ON-times. Unfortunately, these models can be difficult to work with in a queueing theoretic context.

In this paper, we propose an energy cross product $\mathcal{E}$ derived from the sequence of cumulative sums of a video sequence as a single parameter that characterizes VBR video sequences. We then develop a modeling approach that is based on traditional Markov Modulated Rate Processes(MMRP) yet yields processes exhibiting long-range dependent behavior over sets of time scales of practical *interest. Outputs generated by the model are completely determined by $\mathcal{E}$ (together with the mean). Preliminary results show that traffic generated by these models encounters mean delays that are strictly increasing with $\mathcal{E}$ when passing through FIFO $\cdot/D/1$ queues. Therefore, this energy cross product is a good measure of the burstiness of a video sequence and can aid in the development of control strategies for networks with real-time video traffic.*

1 Introduction

Streams of compressed digital video are expected to be a large fraction of the traffic carried on future networks. This presents a significant challenge in the design of these networks, because these streams exhibit characteristics which make their effects on a network difficult to determine. The most critical of these characteristics appears to be the self-similar behavior that was found in empirical studies of single streams of compressed video [1][2][3][4]. This self-similarity is caused by long-term dependence in the data sequences, which means that that the autocorrelation function decays according to a power law instead of exponentially.

Long-term dependence has been shown to have great effects in network performance. For example, Park, et. al shows that the distribution of the file size in an NFS file system has a significant impact on network measurements such as mean buffer size [5]

[6]. Erramilli, et. al obtained similar results for packet traffic with long-term dependence [7]. Therefore, in modeling video sources, long-term dependence must be carefully studied and must appear in artificial traffic generated by the models.

The degree of self-similarity in a time series is measured by the Hurst parameter H. If H is near 0.5, then the sequence is not self-similar, and traditional models, such as MMPP or Poisson processes, may be adequate. As H increases toward 1, the degree of self-similarity increases and the sequence exhibits long-term dependence.

Estimates of the Hurst parameter for actual video sequences are based on the rescaled adjusted range (R/S). These estimates have exceeded 0.7 for almost all of the video sequences that we have examined, thus indicating the presence of long-term dependence. It is thus critical to develop models of these sequences that are "good" in that they capture long-term dependence yet are still tractable in a queueing theoretic sense. Unfortunately, current models appear to fail on either one or the other of these criteria.

Current tractable models of video sources include Markov-Modulated Rate Processes (MMRP), correlated two-dimensional Markov processes, and motion- classified autoregressive models [8]. The MMRP model has been shown to be capable of predicting queue performance distribution with a high degree of accuracy for systems with a number of ON-OFF sources for short video sequences [9]. In this model, a single VBR coded video source is represented as an aggregation of several different types of ON-OFF sources: n_1 sources of type 1, n_2 sources of type 2, etc. For example, an VBR video source can be represented by an ON-OFF sources of one type and, say, ten ON-OFF sources of another type. The parameters of the Markov chain and the arrival rates of each traffic type are obtained by matching moments and other quantities (e.g., index of dispersion) of the outputs to those of the measured traffic [10]. The autocorrelation function of the MMRP processes decays exponentially because the autocorrelation function of each source component does so. Therefore, MMRP-based models cannot be self-similar in a strict theoretical sense.

Models which exhibit strict self-similar behavior can be constructed from fractional ARIMA processes [1], nonlinear chaotic maps [11] , or aggregations of ON-OFF sources with heavy-tailed distributed ON times [12][13]. Unfortunately, none of these models has yet proven to be tractable when used for the analysis of video streams passing through an ATM network.

It might thus appear that one must choose between models which are analytically tractable yet do not exhibit self-similarity, or models which exhibit strict self-similarity yet are not tractable. The way out of this dilemma is to realize that any video sequence is finite in length and that it is possible to create processes which exhibit self-similarity over a range of time scales of practical interest, even though they are not self-similar over all time scales and are thus not strictly self-similar. The goal is to then develop models with simple characterizations that exhibit the long-range dependence required to generate self-similar behavior over the time scales of interest.

In this paper we attempt to achieve this goal by using a single parameter we call the energy cross product $\mathcal{E}$ to characterize long-term dependence in VBR video sequences. While this single parameter cannot capture all characteristics of VBR video sources, it appears to be very important in the analysis, over time scales of interest, of queues with VBR video inputs.

sequence	Hurst Parameter
raceI	0.92
lambI	0.81
MrBeI	0.78
tak2I	0.74

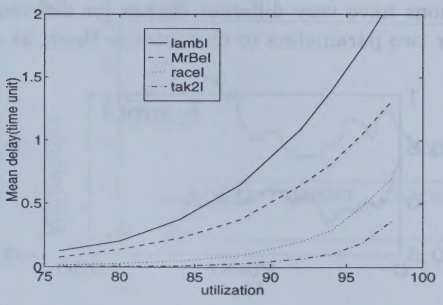


Figure 1: Hurst parameter of several video sequences and mean queueing delay from simulation

2 Parameters for VBR sources

To determine whether various parameters are "good" or "bad" at characterizing video traffic, we fed real video sequences (normalized to a common mean) to a FIFO queue with constant service time and measured the resulting mean delay. Then we tried to relate these measurements to the parameter under study. The most important criterion is that a plot of mean delay versus the parameter must be monotonic and have as high a resolution as possible. Other criteria the parameter must satisfy are that it is easy to estimate from traffic measurements, that it alone specifies a traffic model, and that it has physical meaning.

Several well-known parameters are obvious candidates for characterizing video sequences. However, our analysis indicates none of them is "good" in terms of the criteria just discussed:

2.1 Hurst parameter

The Hurst parameter is a measure of self-similarity in a time series. Figure 1 shows the Hurst parameter of several video sequences and the the mean delay when the individual sequences pass through a FIFO queue. It appears that there is no strong correlation between the *Hurst* parameter and the queue performance. For example, the sequence *raceI* has the highest *Hurst* parameter, which implies strong long-term dependence in the sequence. However, it has very low mean delay. Therefore, the *Hurst* parameter, or any function of it, is not a key parameter of a video sequence as far as mean queueing performance is concerned.

2.2 Autocorrelation function

The *whole* autocorrelation function of a VBR sequence obviously captures the dependence between values in different locations. Recent studies have related the autocorrelation function directly to the performance of a queue [14]. However, since the autocorrelation

functions have very different shapes for different video sequences, it is difficult to find one or two parameters to characterize them, as evident in Figure 2.

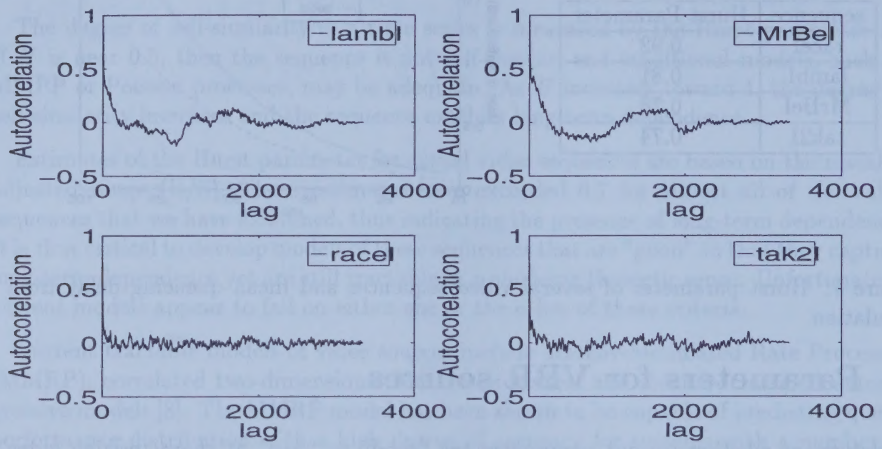


Figure 2: Autocorrelation function of four VBR video sequences. It appears to be difficult to characterize them by one or two parameters.

3 Cumulative processes and a decomposition

In this section we show that a VBR sources can indeed be decomposed into a sum of MMRP processes plus one uncorrelated process. The key is to study the cumulative process (the sequence of cumulative sums) rather than the process itself. Let $\{x(n) : n = 0, 1, 2, ..., N-1\}$ be the frame-size sequence of a video stream and assume it is wide sense stationary. Define the cumulative process $\{s(n) : n = 0, 1, 2, ..., N-1\}$ as $s(n) = \sum_{k=0}^{n}(x_k - \bar{x})$ where $\bar{x}$ is the mean of x. Also define the (complex) random sequences $A_x(n) = \sum_{k=0}^{N-1} x(k)W^{kn}$ and $A_s(n) = \sum_{k=0}^{N-1} s(k)W^{kn}$ where $W = e^{-2\pi\imath/N}$. These sequences are the DFTs of $x(n)$ and $s(n)$, respectively.

Figure 3 shows a VBR video sequence, its cumulative process and their DFTs. Note that the magnitudes of the DFT coefficients of the cumulative process decay much faster than those of the original sequence.

Define $r_M(n) = s(n) - (1/N)\sum_{k=1}^{M} A(k)W^{-kn}$ to be the residual sequence that is left after the first M harmonics are removed from $s(n)$. The Hurst parameter H of the residual sequence is very close to 0.5 for $M = 16$ for all the video sequences we have tested. As M increases, H converges to 0.5. Therefore, the cumulative process can be decomposed with small error into a sum of M sinusoidal sequences plus an uncorrelated sequence.

This decomposition into harmonics leads to an explicit stochastic model for video traffic if we approximate the sinusoidal harmonics of $s(n)$ by triangular harmonics. This simple procedure, which is illustrated in Figure 4(a), is extremely important because each triangular harmonic of $s(n)$ corresponds to a single ON-OFF process of appropriate

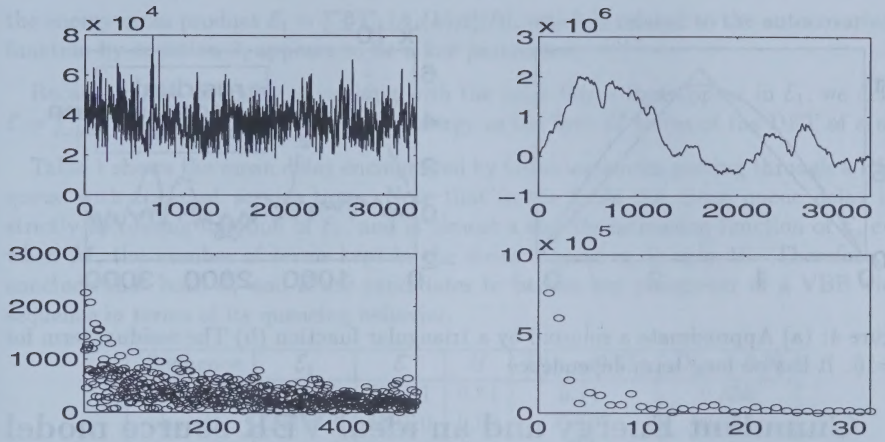


Figure 3: (a) Original sequence $x(n)$. (b) Cumulative sequence $s(n)$. (c) Magnitude of coefficients of the DFT of $x(n)$. (c) Magnitude of coefficients of the DFT of $s(n)$. Note that the DFT of $s(n)$ decays much faster than the DFT of $x(n)$.

"period" in the video sequence $x(n)$. Since ON-OFF processes are the building blocks of MMRP models, the DFT of $s(n)$ points the way to an explicit MMRP model of video sequences as aggregates of ON-OFF sequences and an uncorrelated sequence.

When the above approximation is used, the residual process does no longer approaches zero as the number of terms, M, in the approximation increases. This is not important. The important point is that the residual becomes more and more *uncorrelated* as the number of terms increases. The procedure to decompose the cumulative process is outlined as follows:

(1) Find the DFT coefficient of the harmonic with period $T = N$ where N is the length of the sequence. (2) Approximate the sinusoid by a triangle. (3) Compute the residual (4) Test the residual process to see if it is uncorrelated (Hurst parameter is close 0.5). If not, replace the period by $T/2$ and goto (1). If yes, stop.

For all video sequences we have tested, the above procedure stops after several iterations and the residual process exhibits no long-term dependence. As the number of terms increases, the Hurst parameter does converge to 0.5. In our study, we found that $M = 16$ is large enough to obtain uncorrelated residuals for 30 minute video sequences.

Another key finding is that the magnitudes of the DFT coefficients of the cumulative process for almost all video sequences we have tested are roughly inversely proportional to the frequency. This effect can be seen in Figure 3. This observation gives is the basis of our development, in the next section, of a single parameter that characterizes video sequences.

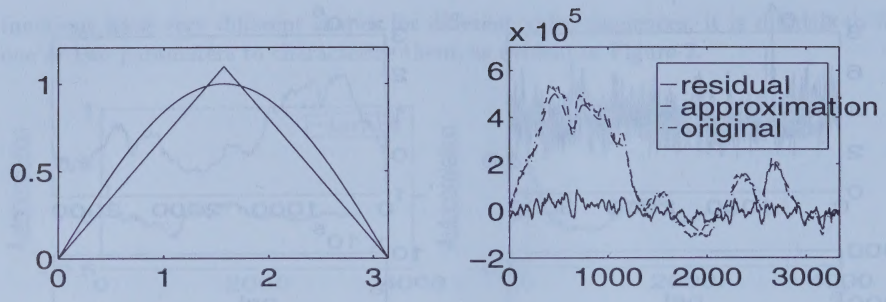


Figure 4: (a) Approximate a sinusoid by a triangular function (b) The residual term for M=16. It has no long term dependence

4 Cumulant Energy and an ideal VBR source model

In this section, we propose a single parameter, the energy cross product, for the classification and stochastic modeling VBR video sequences.

4.1 Cumulant Energy

Let $\{x(n) : n = 0, 1, 2, ..., N-1\}$ be the frame size of a video sequence, $s(n)$ be its cumulative process and A_x be the DFT of x as defined in the previous section. The autocovariance function R_x of $x(n)$ is given by

$$\begin{aligned} R_x(m) &= E[(x(n) - \bar{x})(x(n+m) - \bar{x})] \\ &= E[(s(n) - s(n-1))(s(n+m) - s(n+m-1))] \\ &= R_s(n, n+m) - R_s(n, n+m-1) - R_s(n-1, n+m) \\ &\quad + R_s(n-1, n+m-1) \end{aligned} \tag{1}$$

Given the DFT representation of $s(n)$, namely $s(n) = 1/N \sum_{k=0}^{N-1} A_s(k) W^{-nk}$, we have:

$$\begin{aligned} R_s(n, n+m) &= E[s(n)s(n+m)] = s(n)s(n+m) \\ &= \frac{1}{N^2} \left(\sum_{k=0}^{N-1} A_s(k) W^{-nk} \right) \left(\sum_{l=0}^{N-1} A_s^*(l) W^{(n+m)l} \right) \\ &= \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} A_s(k)\ A_s^*(l) B(k, l, n, n+m), \end{aligned} \tag{2}$$

where $B(k, l, n, n+m) = W^{-nk} W^{(n+m)l}$. Then,

$$\begin{aligned} R_x(n, n+m) &= \frac{1}{N^2} \sum_{k=0}^{N-1} \sum_{l=0}^{N-1} A_s(k) A_s^*(l)\ \{B(k, l, n, n+m) - B(k, l, n, n+m-1) \\ &\quad - B(k, l, n-1, n+m) + B(k, l, n-1, n+m-1)\} \end{aligned} \tag{3}$$

Recent studies have shown that the autocovariances (or autocorrelation) $R_x(n, n+m)$ of the arrival process $x(n)$ plays an important role in queue performance [14]. Therefore,

the energy cross product $\mathcal{E}_1 = \sum_k \sum_l |A_s(k)A_s^*(l)|$, which is related to the autocovariance function by equation 3, appears to be a key parameter.

Because of the difficulty of working with the cross terms that appear in $\mathcal{E}_1$, we define $\mathcal{E} = \sum_{k=1}^{M} A_s(k)A_s^*(k)/\bar{x}^2$, which is the energy in the first M terms of the DFT of $s(n)$.

Table 1 shows the mean delay encountered by video sequences passing through a FIFO queue with constant service time. Note that in the table the mean queue delay is a strictly increasing function of $\mathcal{E}_1$, and is almost a strictly increasing function of $\mathcal{E}$, even when M, the number of terms kept in the series expansion, is only 16. Therefore, we conclude that both $\mathcal{E}_1$ and $\mathcal{E}$ are candidates to be the key parameter of a VBR video sequence in terms of its queueing behavior.

Sequence	$\mathcal{E}_1$	$\mathcal{E}$	H	$\bar{D}(\rho = 0.75)$	$\bar{D}(\rho = 0.9)$
starI	32142.1	8971.1	0.84	0.141	0.856
bondI	24499.4	5391.0	0.87	0.107	0.964
mtv2I	23020.9	3718.8	0.85	0.394	1.027
mtv1I	18049.5	3041.3	0.87	0.127	0.781
lambI	15290.5	3241.4	0.81	0.204	1.090
MrBeI	14855.8	2739.4	0.78	0.139	0.647
asteI	14394.2	5347.3	0.89	0.130	0.437
simpI	11121.7	1977.6	0.81	0.140	0.398
soc2I	10092.2	2410.0	0.92	0.070	0.510
new2I	7203.0	809.3	0.85	0.036	0.297
raceI	5400.1	945.4	0.92	0.028	0.204
soc1I	5290.0	883.5	0.88	0.038	0.267
atppI	5043.6	746.3	0.84	0.011	0.128
sbowI	4785.3	771.5	0.87	0.022	0.153
tak2I	3865.2	728.8	0.74	0.005	0.067
termI	3822.8	741.7	0.80	0.008	0.104
tak1I	2955.6	485.2	0.81	0.001	0.077

Table 1: $\mathcal{E}_1$, $\mathcal{E}$, Hurst parameter and mean dealy $\bar{D}$

4.2 Ideal VBR video source model

An ideal VBR video source model is characterized by two parameters: the mean frame size $\bar{x}$ and the energy cross product $\mathcal{E}$. The output of the model is the sum of M independent Markov Modulated Rate Process(MMRP) x_1, x_2, ..., x_M, plus an uncorrelated residual process. In our study, we found $M = 16$ is sufficient to obtain reasonable approximations for 30 minute video sequences. For the h^{th} MMRP process x_h, the mean ON and OFF time is $T/2h$. When in the ON period, the process generates constant output with rate $\alpha_h = \mathcal{E}^{1/2}/C$ where C is a constant to be found later. While in OFF period, the process generates no output.

The constant C is chosen such that the estimate $\hat{\mathcal{E}}$ of $\mathcal{E}$ is unbiased. The following derivations lead to the appropriate choice of C:

$$\bar{x}^2\hat{\mathcal{E}} = \sum_{k=1}^{M} A_s(k)A_s^*(k) = \sum_{k=1}^{M} \left(\sum_{i=0}^{N-1} (s(i)W^{ki})\right) \left(\sum_{j=0}^{N-1} (s(j)W^{-kj})\right)$$

$$= \sum_{k=1}^{M} \left(\sum_{i=0}^{N-1} \sum_{u=0}^{i} (x(i) - \bar{x}) W^{ki}\right) \left(\sum_{j=0}^{N-1} \sum_{v=0}^{j} (x(v) - \bar{x}) W^{-kj}\right)$$

$$= \sum_{k=1}^{M} \sum_{u=0}^{N-1} \sum_{j=0}^{N-1} \sum_{u=0}^{i} \sum_{v=0}^{j} (x(u) - \bar{x})(x(v) - \bar{x})\, W^{ki-kj} \tag{4}$$

$$\bar{x}^2 E[\hat{\mathcal{E}}] = E[\sum_{k=1}^{M} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sum_{u=0}^{i} \sum_{v=0}^{j} (x(u) - \bar{x})(x(v) - \bar{x}) W^{ki-kj}$$

$$= \sum_{k=1}^{M} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sum_{u=0}^{i} \sum_{v=0}^{j} R_x(u,v) W^{ki-kj} \tag{5}$$

Now we need to find the autocorrelation function R_x for sums of MMRP processes. For the h^{th} MMRP process x_h, the autocorrelation function is given by $R_{x_h}(u,v) = (1-2h/N)^{|u-v|}(\alpha_h/2)^2$. The autocorrelation function of the uncorrelated residual process is zero.

$$R_x(u,v) = \sum_{h=1}^{M} R_{x_h}(u,v) = \sum_{h=1}^{M} (1 - \frac{2h}{N})^{|u-v|} \frac{{\alpha_h}^2}{4}$$

$$= \frac{\mathcal{E}}{4C^2} \sum_{h=1}^{M} (1 - \frac{2h}{N})^{|u-v|} \tag{6}$$

Therefore,

$$\bar{x}^2 E[\hat{\mathcal{E}}] = \frac{\mathcal{E}}{4C^2} \sum_{k=1}^{M} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sum_{u=0}^{i} \sum_{v=0}^{j} W^{ki-kj} \left[\sum_{h=1}^{M} (1 - \frac{2h}{N})^{|u-v|}\right] \tag{7}$$

From this last equation it is clear that we should choose C such that

$$C^2 = \sum_{k=1}^{M} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} \sum_{u=0}^{i} \sum_{v=0}^{j} W^{ki-kj} \sum_{h=1}^{M} (1 - \frac{2h}{N})^{|u-v|} / 4\bar{x}^2, \tag{8}$$

for then $E[\hat{\mathcal{E}}]$ is an unbiased estimate of $\mathcal{E}$.

Preliminary simulation results show that queueing behavior is very strongly related to the energy cross product $\mathcal{E}$. We are currently doing the analysis to precisely determine this link.

5 Conclusion and future work

MMRP models have previously been shown to be able to predict queue performance with a high degree of accuracy for systems with a number of ON-OFF sources. The eigenvalues for the spectral expansion of the queue length distribution for queues with MMRP inputs can be obtained using an iteration algorithm in which the complexity does not grow as fast as the number of states. Therefore MMRP models are analytically tractable.

This is significant because we have shown in this paper how MMRP models can also accurately model video sequences and how they can capture the dependence structure of these processes on time scales of practical interest. We have shown that a single parameter, the energy cross product $\mathcal{E}$, completely specifies an MMRP-based model that accurately models video sequences.

The approach in this paper thus shows significant promise of being both analytically tractable while still accurately accounting for long-term dependent behavior. It could become a significant tool for the design of high-speed networks carry video traffic.

6 Acknowledgment

The authors would like to thank O. Rose of University of Würzbury, Germany, for providing the MPEG traces used in this paper. They are available from ftp://ftp-info3.informatik.uni-wuerzburg.de/pub/MPEG/.

References

[1] Mark W. Garrett and Walter Willinger. Analysis, modeling and generation of self-similar VBR video traffic. *SIGCOMM*, pages 269–280, 1994.

[2] Jan Beran, Robert Sherman, Murad S. Taqqu, and Walter Willinger. Long-range dependence in variable-bit-rate video traffic. *IEEE Transactions on Communications*, 43(2-4):1566–1579, Feb.-Apr. 1995.

[3] Ashok Erramilli and Walter Willinger. Fractal properties in packet traffic measurements. *Bellcore Technical Report*, 1990.

[4] O. Rose. Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems. Technical report, Institute of Computer Science, University of Würzburg, Germany, 1995.

[5] Kihong Park, Gitae Kim, and Mark Crovella. On the relationship between file sizes, transport protocols, and self-similar network traffic. In *International Conference on Network, Protocols*, October 1996.

[6] Kihong Park, Gitae Kim, and Mark Crovella. On the effect of traffic self-similarity on network performance. Submitted for publication, 1996.

[7] Ashok Erramilli, Onuttom Narayan, and Walter Willinger. Experimental queueing analysis with long-range dependent packet traffic. *IEEE/ACM Transactions on Networking*, 4(2):109–223, April 1996.

[8] Basil Maglaris, Dimitris Anastassiou, Prodip Sen Gunnar Karlsson, and John Robbins. Performance models of statistical multiplexing in packet video communications. *IEEE Transactions on Communications*, 36(7):834–843, Jul. 1988.

[9] A. Baiocchi, N. B. Melazzi, M. Listanti, A. Roveri, and R. Winkler. Loss performance analysis of an ATM multiplexer loaded with high-speed ON-OFF sources. *IEEE Journal on Selected Areas in Communication*, 9(3):388–393, April 1991.

[10] Hyong W. Lee and Jon W. Mark. ATM network traffic characterization using two types of On-Off sources. In *IEEE INFOCOM*, pages 152–159, 1993.

[11] A. Erramilli and W. Willinger. An application of deterministic chaotic maps to characterize packet traffic. In *Proc. 7th ITC Specialists Seminar*, pages 8.1.1–8.1.3, 1990.

[12] Walter Willinger, Murad S. Taqqu, Will E. Leland, and Danial V. Wilson. Self-similarity in high-speed packet traffic: Analysis and modeling of Ethernet traffic measurements. *Statistical Science*, 10(1):67–85, 1995.

[13] W. Willinger, M.S. Taqqu, R. Sherman, and D.V. Wilson. Self-similarity through high-variability: Statistical analysis of Ethernet LAN traffic at the source level. In *ACM/SIGCOMM*, 1995.

[14] Predrag R. Jelenvović and Aurel A. Lazar. A network multiplexer with mltiple time scale and subexponential arrivals. In *Lecture Notes in Statistics: Stochastic Networks*, pages 215–233. Springer, 1996.

[15] Harry Heffes and David M. Lucantoni. A markov modulated characterization of packetised voice and data traffic and related statistical multiplexer performance. *Journal on Selected Area in communications*, sac-4(6):856–867, Sep. 1986.

Fractional Brownian Motions and Lévy Motions as Limits of Stochastic Traffic Models

TAKIS KONSTANTOPOULOS* AND SI-JIAN LIN†
Department of Electrical and Computer Engineering
University of Texas at Austin‡

October 1996

ABSTRACT

This article offers an overview of some approximating schemes and their performance analysis for communication networks driven by traffic processes possessing long-range dependence. The schemes lead to the so-called *fractional Brownian motion models* or *Lévy motion models*. Under certain conditions, the load of a network with bursty inputs can be mathematically well approximated by a reflected multi-dimensional Lévy motion with certain drift. Although fractional Brownian motion models are mathematically fundamental in describing long-range dependence and self-similarity, it is argued that a particular type of Lévy models is a good candidate for explaining phenomena observed in networking practice. Unlike fractional Brownian motions, Lévy motions are semimartingales, and hence stochastic calculus plays an important role in investigating the stationary regime and quality of service of the network. We point out that there is a systematic way for writing evolution equations and equations for the stationary regime. En passant, we deal with questions of existence and uniqueness of a stationary regime and the question of existence and construction of a suitable topology that makes the multi-dimensional reflection mapping (which is responsible for the precise characterization of the limiting evolution equations) continuous, thereby allowing us to pass the limit inside the reflection operator.

1 Approximation schemes for multiplexed source traffic

Let us start by introducing the traffic of individual source models. It is a common scenario (Willinger et al. (1995)) that the required services for network sessions experience high variability (imagine the burstiness of `real-time video` traffic, `ftp` traffic, or data transfer via the

*This work was supported in part by NSF Faculty Career Development Award NCR 95-02582, and by Grant ARP 224 of the Texas Higher Education Coordinating Board

†This article summarizes part of the author's dissertation. It is supposed to be an overview of results whose proofs can be found in a couple of forthcoming papers

‡Authors' e-mail addresses: takis@alea.ece.utexas.edu, sjlin@alea.ece.utexas.edu

WWW). Hence it is reasonable to consider a large quantity of independent packet trains of ON/OFF sources, whose ON (and possibly OFF) periods have heavy-tailed distributions (e.g. the Pareto distribution). Clearly, we can think of the packet train as a 0-1 valued process $\{X_t\}$, where 1 is assigned to the ON periods and 0 elsewhere. Physically, X_t is the instantaneous rate at time t. Assume that the lengths of the ON periods are i.i.d. distributed like τ. The lengths of the OFF periods are also i.i.d., distributed like θ, and independent of the ON periods. Certainly we require $E[\tau + \theta] < \infty$ to ensure that each source model has a stationary regime. Now we are in a position to superimpose a large number of independent copies of $\{X_t\}$. Two approximation schemes apply. They are described in the sequel. The first one is part of earlier work, but is described for completeness. The second one is new. Due to space limitations, explicit proofs are omitted.

1.1 The Gaussian approximation scheme

The first one, which we call *Gaussian approximation*, due to the Gaussian central limit theorems, considers i.i.d. copies of $\{X_t\}$, denoted by $\{X_t^k\}_{k=1}^M$. We assume that M is the (large) number of sources to be multiplexed and consider the total rate $\sum_{k=1}^M X_t^k$. The large summation results in a asymptotically Gaussian process after centering and suitable norming since each X^k is a 0-1 process. Let us clarify this using the simplified model of Taqqu and Levy (1986). To illustrate $\{X_t\}$, we assume that

$$\ldots < T_{-1} < T_0 \leq 0 < T_1 < T_2 < \ldots$$

is a *stationary* point process, and $\{T_n - T_{n-1}, n \neq 1\}$ are i.i.d. integer-valued r.v.'s distributed as, say, ν. The interval $T_1 - T_0$ containing 0 is independent of $\{T_n - T_{n-1}, n \neq 1\}$, and, from standard renewal theory, the stationarity of the point process $\{T_n\}$ is equivalent to requiring

$$P[T_1 > n] = \frac{1}{E\nu} \sum_{u=n+1}^{\infty} P[\nu \geq u] \,. \tag{1}$$

Independently, let $\{W_n\}$ be i.i.d. with Bernoulli 0-1 distribution with parameter $p = 1/2$. Now the process $\{X_t\}$ is defined by

$$X_t := \sum_{n=-\infty}^{+\infty} W_n \, 1(T_n \leq t < T_{n+1}) \,.$$

Clearly it is an ON/OFF process. It can be shown that if ν has the Pareto-type distribution (thus exemplifying the heavy-tail assumption), i.e. [1]

$$P[\nu > x] \sim \kappa x^{-\alpha} \quad (2 > \alpha > 1) \tag{2}$$

[1] The symbol $a_x \sim b_x$ means that $a_x/b_x \to 1$, as $x \to \infty$, whereas the symbol $a_x \approx b_x$ means that a_x/b_x converges to some unspecified constant.

for large x, then the ON and OFF periods also have the same type of tails. The *centered superimposed* process, $\{Y_t^M\}$, is defined by

$$Y_t^M := \frac{1}{\sqrt{M}} \sum_{k=1}^{M} (X_t^k - EX_t^k) \ , \quad t = 0, 1, 2, \ldots$$

Then the autocovariance of $\{Y_t^M\}$ is computed via (1):

$$\mathrm{cov}(Y_{t+n}^M, Y_t^M) = E[(X_0 - 0.5)^2, T_1 > n] = 0.25\, P[T_1 > n] \approx n^{1-\alpha} \ .$$

From the standard central limit theorem, it is readily seen that $\{Y_t^M\}$ converges (in the sense of finite dimensional distributions), as M goes to infinity, to a centered stationary Gaussian process $\{Y_t\}$, which has lag-n covariance $EY_tY_{t+n} \approx n^{1-\alpha}$ for large n. It is said that it possesses *long-range dependence (LRD)* since the covariances are not summable over n. This approximation scheme leads to a fractional Brownian motion (FBM) in the sense that the process

$$Z_s^n := \frac{1}{n^H} \sum_{t=1}^{[ns]} Y_t \,, \tag{3}$$

where $H := (3 - \alpha)/2$, converges weakly as $n \to \infty$ to a fractional Brownian motion under the locally uniform topology. The proof of the weak convergence in the Skorokhod space follows from the general approximation schemes of Konstantopoulos and Lin (1995).

1.2 The Poisson approximation scheme

On the other hand, empirical studies on some very bursty traffic (notably the reports of Garrett and Willinger (1994) on `VBR Video`) show that *non-Gaussian laws* should play an important role in modeling statistically self-similar traffic. In Garrett and Willinger (1994), the time series of a two hour video traffic—using the by now well known movie source `Star Wars` with discrete cosine transform and Huffman coding—is studied, resulting in a "hybrid Gamma/Pareto" distribution. This distribution is *non-symmetric* around the mean: it has Pareto-type *heavy tail toward the positive infinity*, and *very thin tail toward the negative infinity*. We believe that this distribution actually approximates a skewed stable law (see the recent monograph of Samorodnitsky and Taqqu (1994) which contains the relevant concepts).

Indeed, *Lévy motions* are limits of the multiplexed ON/OFF traffic with heavy-tailed ON periods, via an approximation scheme called *Poisson approximation* (Lin (1996)). We again denote by $\{X_t\}$ the ON/OFF packet train with the distribution of ON periods and OFF periods being τ, θ respectively, and $E[\tau + \theta] < \infty$ is assumed to ensure stationarity. Now, when a large number of sources is being multiplexed, the total rate of transmission is bounded due to the consideration of limited bandwidth in practice. This is different from the Gaussian approximation, where infinite i.i.d. copies of $\{X_t\}$ are used to represent source models, resulting in a linear growth of mean traffic load when the number of sources is increased. The idea of

keeping the total transmission rate bounded can be realized mathematically as follows. With M denoting the number of sources, construct M i.i.d. 0-1 processes $\{X_t^{M,i}, 1 \le i \le M\}$, which are similar to $\{X_t\}$, except that we keep the distribution of the ON period, τ, independent of M, while that of the OFF period, θ_M, depends on M. The total transmission rate is then $\lambda_M := M(E[\tau + \theta_M])^{-1}$. To obtain a meaningful approximation, we need λ_M to approach to some $\lambda < \infty$. This translates to a condition on the mean of θ_M, i.e. that $E\theta_M \sim \lambda^{-1}M$. We would like to utilize the superposition theorem for rare point processes. To this end, we think of the processes $\{X_t^{M,i}\}$ as marked point processes: the points are the time epochs when the ON periods start, and the marks are the lengths of the corresponding ON periods. Hence the superposition is imposed upon the points, and under the assumption of $\lambda_M \to \lambda$, the Palm-Khintchine theorem (a standard result in queueing theory as explained, for example, in Gnedenko and Kovalenko (1989)) gives the stationary Poisson point process with intensity λ as the limiting process. The marks remain as i.i.d. r.v.'s with distribution τ, representing the duration of the transmissions. In summary, under the condition that the total transmission rate λ_M converges to $\lambda < \infty$, the multiplexed traffic can be approximated by a stationary Poisson shot-noise model (Likhanov et al. (1995)): the underlying point process is a stationary Poisson process $\{T_n\}$ with intensity λ, and the "shot-noise" is an i.i.d. sequence of r.v.'s $\{W_n\}$ with $W_n \stackrel{\mathcal{D}}{=} \tau$, also independent of $\{T_n\}$. Finally, the aggregate traffic Y_t^* is defined as the number of alive sessions at time t:

$$Y_t^* := \sum_{n=-\infty}^{\infty} 1(T_n \le t < T_n + W_n) \; . \tag{4}$$

We shall consider the centered process of $\{Y_t^*\}$, i.e., $\{Y_t^* - \lambda E\tau\}$, and denote it by $\{Y_t\}$. The equality $EY_t^* = \lambda E\tau$ is from Campbell's formula (see Baccelli and Brémaud (1994)).

This process Y_t is examined by Likhanov et al. (1995) by studying the second-order statistics (although the authors considered the discrete time case; the continuous time case is just a convenient (for us) extension). The result can be phrased as: assuming the presence of the heavy-tailed Pareto type distribution of τ, i.e., $P[\tau > x] \approx x^{-\alpha}$ for some $1 < \alpha < 2$, then, up to second-order statistics and appropriate time rescaling, the centered process $\{Y_t\}$ is approximately a fractional Brownian noise. Precisely, define the sequence of processes $\{Z_s^n\}$:

$$Z_s^n = \frac{1}{n^H} \int_0^{ns} Y_t \, dt \; , \qquad \text{with} \quad H = \tfrac{3-\alpha}{2} \; . \tag{5}$$

Then, we have $\mathrm{cov}(Z_s^n, Z_t^n) \to \mathrm{cov}(B_t^H, B_s^H)$ for a FBM B^H.

However, the queueing analysis of the process Y (Likhanov et al. (1995)) shows that the loss probability is very different from the FBM setup, suggesting a peculiar behavior of the new model. It turns out that the processes $\{Z^n\}$ defined by (5) converge to the *zero-valued* process under the locally uniform topology of the path space $D[0, \infty)$. This means the exponent H is too big as a norming factor, although (5) stabilizes the variance. In order to obtain a sensible

limit result, we shall redefine $\{Z^n\}$ of (5) by

$$Z_s^x := \frac{1}{x^{1/\alpha}} \int_0^{xs} Y_t \, dt \ , \qquad x, s \in \mathbf{R}_+ \ . \tag{6}$$

Notice that $\{Z_s^x\}$ parameterized by x is a *family* of processes rather than just a sequence. This is more natural when continuous time processes are concerned, and it is more general. Also, the new exponent, $1/\alpha$, is between $1/2$ and 1, and strictly less than $H = (3-\alpha)/2$ when $\alpha \in (1,2)$, which reflects the fact that H is indeed too large for scaling.

Theorem 1 *Under the heavy-tail assumption (2) on the session durations, the aggregate traffic* $\{Y_t^*\}$ *given in (4) is approximately an* α*-stable Lévy noise perturbation around a constant rate transmission. Precisely, in the sense of convergence of finite dimensional distribution (f.d.d.), the processes* $\{Z^x\}$ *defined in (6) converge to an* α*-stable Lévy motion* L *with* $L_0 = 0$ *and* $L_1 \sim S_\alpha(\sigma, 1, 0)$*, where the constant*

$$\sigma := \left(\frac{\kappa \lambda \Gamma(2-\alpha)}{\alpha - 1} \left[-\cos \frac{\pi \alpha}{2} \right] \right)^{1/\alpha} ,$$

Γ *is the Gamma function,* κ *and* α *are as in (2), and* λ *is the intensity of the session arrivals.*

Here, an α-stable Lévy motion is a random process with stationary independent increments, and the increment L_1 on a time interval of size 1 is distributed as the *extremely skewed stable law* $S_\alpha(\sigma, 1, 0)$ (Samorodnitsky and Taqqu (1994)). In particular, we have

$$\begin{aligned} P[L_1 > \lambda] &\sim C^1_{\alpha,\sigma} \lambda^{-\alpha} , \text{ as } \lambda \to +\infty , \\ P[L_1 < \lambda] &\sim C^2_{\alpha,\sigma} |\lambda|^{-\alpha/(2(\alpha-1))} \exp(-C^3_{\alpha,\sigma} |\lambda|^{-\alpha/(\alpha-1)}) , \text{ as } \lambda \to -\infty , \end{aligned}$$

where $C^i_{\alpha,\sigma}$, $i = 1, 2, 3$,are positive constants. This tail behavior resembles the one described in Garrett and Willinger (1994), where the distribution is heuristically patched by a heavy tailed Pareto distribution on the positive side and a Gamma distribution on the negative side.

The proof of this theorem involves computation of the limits of the characteristic functions of finite dimensional distributions, using the formula for the characteristic function of a Poisson shot noise (see Takács (1955), for instance). This theorem can be generalized to a larger class of variables with heavy-tails, that is, the Pareto-type assumption for function $P[\nu > x]$ in (2) can be relaxed to include a *regularly varying function.* The full-fledged result and proof is given in Lin (1966).

It is believed that the convergence in Theorem 1 also holds under the Skorokhod M_1 topology, since it is known that it is impossible for a sequence of continuous processes to converge to a process with jumps under the (usual) Skorokhod topology. The Skorokhod M_1 topology is first introduced in the article of Skorokhod (1957), which is a topology on the càdlàg (right-continuous with limits from the left) path space and is weaker than the usual Skorokhod

topology, which is called the J_1 topology by Skorokhod (1957). The Skorokhod M_1 topology uses the parameterization of the graph of a càdlàg path with the jumps connected via straight lines. For details see Skorokhod (1957).

2 Convergence to Lévy networks and performance analysis

We now consider an open network consisting of d servers with routing matrix $P = [p_{ij}]$. Assume that P has spectral radius strictly less than one, so that, in the absence of external inputs, any work in the network will eventually be finished. Although not necessary, we deal with a so-called fluid model, for simplicity. In the case of single server, the process $\{Y_t^*\}$ defined in (4) can be regarded as a workload input; the servers work up to the maximum capacities (work conservation principle). We denote by A_t the input workload process. It is a d-dimensional non-decreasing process, and we assume that A_t is asymptotically Lévy (the exact meaning is declared later). The capacities, or service rates, or maximal throughput rates are recorded as a d-dimensional vector ν. Then we can write the balance equation for the remaining workload Q according to the work conservation principle:

$$\{Q_t\} = \mathcal{R}\{X_t\}\,,$$

where $\mathcal{R}$ is the reflection mapping in $\mathbf{R}_+^d$ with the reflection matrix $R = (I - P)'$, and $X_t = (X_t^i, i = 1, \ldots, d)$ is a d-vector with components

$$X_t^i := Q_0^i + A_t^i + \sum_{j=1}^{d} \nu_j p_{ji} t - \nu_i t\,. \tag{7}$$

(Note that A' denotes the transpose of a matrix A). Let us briefly explain why the reflection mapping is involved. Imagine some extra work on each server which keeps the servers working at their full capacities. The work conservation principle is equivalent to assigning low priority to the imaginary work. Denote by Y_t the amount of work servers have done on the imaginary work up to time t. Then, for the remaining "real" workload Q_t, we have

$$0 \le Q_t = Q_0 + A_t + (P'(\nu t - Y_t)) - (\nu t - Y_t) = (Q_0 + A_t - R\nu t) + RY_t\,.$$

This equation or rather inequality has *the* minimal solution for Y (the so-called reflector, denoted by $Y = \mathcal{L}X$), where X is given in (7). Since the imaginary work has low priority, we conclude that $Y = \mathcal{L}X$, and hence $Q = \mathcal{R}X$.

2.1 Network approximations: spatial limits

To pass on to the limit, we keep the network topology intact and assume the external workload vector A and service rate vector ν vary with an extra index N. The assumption of Lévy

approximation on A or A^N is defined explicitly by

$$N^{-1/\alpha}(A^N_{Nt} - \lambda^N Nt) \Rightarrow L_t, , \tag{8}$$

under the M_1 topology, where $\lambda^N := EA^N_1$ is the input rate, L a zero-mean d-dimensional α-stable Lévy motion with $\alpha > 1$. We now introduce the natural scaling assumptions, which are similar to those of Reiman (1984) (the Brownian network case) and to those of Konstantopoulos and Lin (1996) (the fractional Brownian case):

H1 $N^{1-1/\alpha}[\lambda^N + (P' - I)\nu^N] \to -\mu \in \mathbf{R}^d$,

H2 Q^N_0 *is independent of* A^N, *and*

H3 $N^{-1/\alpha}Q^N_0 \xrightarrow{P} Q_0$. The next theorem deals with the passage of the limit inside the reflection operator. To justify this passage it was necessary to weaken the M_1 topology because one can devise examples showing that $\mathcal{R} : (D^d[0,\infty), M_1) \to ((D^d[0,\infty), M_1)$ is not continuous when $d > 1$. We construct a topology, weaker than M_1 but stronger than M_2, which we call M_R topology and which makes the map

$$\mathcal{R} : (D^d[0,\infty), M_1) \to ((D^d[0,\infty), M_R)$$

continuous. The reason that it is called M_R is that it depends on the structure of the matrix $R = (I - P)'$. For details see Lin (1996). The main point is the following theorem:

Theorem 2 *Under the conditions H1-H3, the (rescaled) workload processes* $\{N^{-1/\alpha}Q^N_{Nt}\}$ *converges weakly, in the space* $((D^d[0,\infty), M_R)$, *to the reflected process of* $Q_0 + L_t - \mu t$ *in* $\mathbf{R}^d_+$ *with the reflection matrix* R, *where* Q_0 *is independent of* L.

We now write $X_t = Q_0 + L_t - \mu t$, the drifted Lévy motion with initial distribution Q_0, and $Z = \mathcal{R}X$, the reflected process of X in $\mathbf{R}^d_+$ with the reflection matrix R. We shall investigate the stationary regime of Z, i.e., under what kind of initial distribution Q_0, the process Z is stationary.

2.2 Stationarity, stability and performance: temporal limits

It is known that when $d = 1$ (the single server case), the stationary distribution has an explicit form when the drift $-\mu t$ is negative (Reich's formula):

$$Q_0 = \sup_{t \geq 0}(L_t - \mu t) \ . \tag{9}$$

The Lévy motion L of Theorem 1 has skewed stable marginals, and it is shown in Lin (1986) that L admits a càdlàg version with no negative jumps. The supremum of this kind of Lévy processes is well-known (Bingham (1975), Harrison (1977)) via the Laplace transform ($\theta > 0$)

$$Ee^{-\theta Q_0} = \mu/(\mu + C_L\theta^{\alpha-2}) \ , \tag{10}$$

where the constant C_L depends on the process L only. The tail behavior is then obtained by a Tauberian Theorem. In short, we have

$$P[Q_0 \geq x] \approx x^{1-\alpha}.$$

Note that this estimate coincides the one obtained in Likhanov et al. (1995) (a simplified model) and the upper bound of Anantharam (1995), where essentially the process Y^* in (4) is studied instead of the heavy-traffic limit. Also, this tail behavior is sharply different from that of the stationary regime for reflected FBM, which is derived by the large deviation principle by Duffield and O'Connell (1993).

Unfortunately, when $d > 1$, there is no such simple formula (9) for the stationary distribution Q_0 (if it exists!) due to the complexity of reflection mapping on a high dimensional space. At least, in the light of the work by Harrison and Williams (1987), a sufficient and necessary condition for the existence of a stationary regime can be obtained, by carefully exploiting the pure jump structure of the Lévy motion L (Lin (1996)):

Theorem 3 *A stationary regime for the reflected Lévy motion $L_t - \mu t$ exists if and only if all the entries of $R^{-1}\mu$ are strictly positive.*

It is developed in Lin (1986) a new approach for deriving the formula (10), using stochastic calculus for jump processes (see Bichteler (1981), Protter (1990)) and Palm calculus (see Baccelli and Brémaud (1994)) (the Palm calculus is very effective in dealing with the "jump terms" appeared in the Itô's formula). Remarkably, this new approach is applicable to high dimensional case (the network case), and the case when both positive and negative jumps are encountered. Although the stationary distributions are not fully characterizable (this is unfortunately true even for the Brownian models considered by Harrison and Williams (1987)) our method yields interesting results on these distributions. For example, in contrast to the classical queueing theory and the Brownian motion models developed in Harrison and Williams (1987), the following holds:

Theorem 4 *Except for the trivial case, where the network consists of stand-alone servers and all the traffic are local, the Lévy motion models allow no product form for the stationary distributions.*

An intuitive explanation of this theorem is, in a "true" network, very bursty traffic has strong impact on all the network nodes it passes through, causing heavy correlations.

3 Conclusions

We considered two types of approximation schemes for multiplexed ON/OFF source models with heavy-tailed ON periods. Both multiplexed traffic have same type of long-range dependence as far as the second order statistics are concerned. The first approximation yields Gaussian models and fractional Brownian motion. This can be exploited at a network level and this was done in an earlier paper (Konstantopoulos and Lin (1996)). Herein we emphasized the second approximation scheme, called Poisson approximation, which led to non-Gaussian models with skewed stable marginals. We proposed the (reflected) Lévy motions as the choice of models when a non-Gaussian phenomenon is encountered (e.g. as suggested by statistical analyses of empirical distributions). Mathematically, these models are limits of the remaining workload under suitable scaling conditions. The loss probability for one server case was obtained. To deal with the multidimensional case, it was necessary to introduce a new topology in the range of the reflection operator in order to make it continuous. We pointed out that it is possible to apply the tools of stochastic calculus and Palm calculus to investigate further the network behavior of the Lévy models.

References

[1] Anantharam, V. (1995). On the sojourn time of sessions at an ATM buffer with long-range dependent input traffic. *Proc. 34th IEEE CDC*, New Orleans.

[2] Baccelli, F. and Brémaud, P. (1994). *Elements of Queueing Theory.* Springer-Verlag, New York.

[3] Baccelli, F. and Foss, S. (1993). Stability of Jackson-Type Queueing Networks, I. *INRIA Rapport de Recherche* No. 1945. Also in *QUESTA*, Vol. **17**, No. I-II.

[4] Bichteler, K. (1981). Stochastic integration and L^p-theory of semimartingales. *Annals of Probability*, **9**, 49-89.

[5] Bingham, N.H. (1975). Fluctuation theory in continuous time. *Adv. Appl. Prob.*, **7**, 705-766. John Wiley & Sons, New York.

[6] Duffield, N.G., and O'Connell, N. (1993) Large deviations and overflow probabilities for the general single-server queue, with applications. To appear in *Proc. Cambridge Phil. Soc.*.

[7] Garrett, M.W. and Willinger, W. (1994). Analysis, modeling and generation of self-similar VBR video traffic. *ACM SigComm*, London.

[8] Gnedenko, B.V. and Kovalenko, I.N. (1989). *Introduction to Queueing Theory.* Birkhäuser, Boston.

[9] Harrison, J.M. (1977). The supremum distribution of a Lévy process with no negative jumps. *Adv. Appl. Prob.*, **9**, 417-422.

[10] Harrison, J.M. and Williams, R.J. (1987). Brownian Models of open queueing networks with homogeneous customer populations. *Stochastics*, **22**, 77-115.

[11] Konstantopoulos, T. and Lin, S.J. (1995). Approximating fractional Brownian motions. *Techn. Rep. ECE, U.T. Austin.*

[12] Konstantopoulos, T. and Lin, S.J. (1996). Fractional brownian approximations of stochastic networks. In: *Stochastic Networks, Stability and Rare Events*, Lecture Notes in Statistics, **117**, 257-274, Springer-Verlag, New York.

[13] Likhanov, N., Tsybakov B. and Georganas, N.D. (1995). Analysis of an ATM buffer with self-similar ("fractal") input traffic, *Proc. IEEE INFOCOM'95*, Boston.

[14] Lin, S.J. (1996). *Ph.D. Dissertation.* ECE Department, University of Texas at Austin (in preparation).

[15] Protter, P. (1990). *Stochastic Integration and Differential Equations,* Springer-Verlag, New York.

[16] Reiman, M. (1984). Open queueing networks in heavy traffic. *Mathematics of Operation Research*, **9**, 3, 441-458.

[17] Samorodnitsky, G. and Taqqu, M. (1994). *Stable Non-Gaussian Random Processes.* Chapman & Hall, New York.

[18] Skorokhod, A.V. (1957). Limit theorems for stochastic processes. *Theory Probab. Appl.*, **1**, 261-290.

[19] Takács, L. (1955). On secondary processes generated by a Poisson process and their applications in physics. *Acta Math. Acad. Sci. Hungar.*, **5**, 203-236.

[20] Taqqu, M.S. and Levy, J. (1986). Using renewal processes to generate long-range dependence and high variability. In: E. Eberlein and M.S. Taqqu, editors, *Dependence in Probability and Statistics*, 73-89, Boston. Birkhäuser.

[21] Willinger, W., Taqqu, M.S., Sherman, R. and Wilson, D.V. (1995). Self-similarity through high-variability: statistical analysis of Ethernet LAN traffic at the source level. *Proc. of the ACM/Sigcomm'95.*

ASYMPTOTICS FOR M/G/1 LOW-PRIORITY WAITING-TIME TAIL PROBABILITIES

Joseph Abate
AT&T retired
900 Hammond Road
Ridgewood, NJ 07450-2908

Ward Whitt
AT&T Labs
Room 2C-178
Murray Hill, NJ 07974-0636

Abstract

We consider the classical M/G/1 queue with two priority classes and the nonpreemptive and preemptive-resume disciplines. We show that the low-priority steady-state waiting-time can be expressed as a geometric random sum of i.i.d. random variables, just like the M/G/1 FIFO waiting-time distribution. We exploit this structures to determine the asymptotic behavior of the tail probabilities. Unlike the FIFO case, there is routinely a region of the parameters such that the tail probabilities have non-exponential asymptotics. When non-exponential asymptotics holds, the asymptotic form tends to be determined by the asymptotics for the high-priority busy-period distribution.

1. Introduction

Recently there has been great interest in the asymptotic behavior of waiting-time and queue-length tail probabilities, largely motivated by the desire to exploit the efficiencies of statistical multiplexing in emerging high-speed communication networks. The asymptotic results are intended to generate approximations that can be used to help manage the network, e.g., to determine required buffer sizes and connection admission control policies. Since models with infinite waiting room are usually easier to analyze, a steady-state tail probability in a model with infinite waiting room is often used to approximate the steady-state probability of buffer overflow at a switch in the network.

To discuss the asymptotic results, let $W(t)$ be the *cumulative distribution function* (cdf) of the steady-state waiting time and let $W^c(t) \equiv 1 - W(t)$ be the associated *complementary cdf* (ccdf). Then the standard exponential asymptotics can be expressed as

$$W^c(t) \sim \alpha e^{-\eta t} \quad \text{as } t \to \infty \,, \tag{1.1}$$

where η and α are positive constants called the *asymptotic decay rate* and the *asymptotic constant*, respectively, and $f(t) \sim g(t)$ as $t \to \infty$ means that $f(t)/g(t) \to 1$ as $t \to \infty$ [1]. Alternatively, large deviation arguments [12] yield weaker logarithmic asymptotics of the form

$$\log W^c(t) \sim -\eta t \quad \text{as } t \to \infty \,. \tag{1.2}$$

The limit (1.1) suggests the approximation $W^c(t) \approx \alpha e^{-\eta t}$ for suitably large t. The approximation (1.2) suggests the approximation $W^c(t) \approx e^{-\eta t}$. The latter approximation is also natural to use starting from (1.1) if the value of the asymptotic constant is difficult to obtain, which is often the case.

The asymptotics in (1.1) and (1.2) serve as a basis for the notion of effective bandwidths in high-speed networks [5]. Even though the simple approximation $W^c(t) \approx e^{-\eta t}$ and the notion of effective bandwidths based directly on it can perform poorly in the intended setting with statistical multiplexing (because α in (1.1) can be very different

from 1) [6], the approach, with appropriate refinements, remains very appealing for engineering.

However, there is growing concern that traffic burstiness (as evidenced by long-range dependence and self-similarity [14]) may lead to asymptotics of a different form than (1.1) and (1.2). In particular, the tail probabilities may decay more slowly than exponentially, which is referred to as *long-tail* (or heavy-tail) *asymptotics* [7], [11], [13]. For example, $W^c(t)$ might have a *power tail*, i.e.,

$$W^c(t) \sim \alpha t^{-\beta} \quad \text{as } t \to \infty \, . \tag{1.3}$$

The asymptotic analysis described above has been done for queues with the first-in first-out (FIFO) service discipline. However, it is becoming common to design communication networks with different priority classes. Thus, it is natural to investigate corresponding asymptotics for priority queues. This paper describes first steps in that direction.

Here we describe some of the asymptotic results that we have recently obtained (in a much longer paper with the same title) for the low-priority steady-state waiting-time distribution in the M/G/1 queue with two priority classes and the nonpreemptive and preemptive-resume disciplines. It still remains to treat priority queues with more general arrival processes, as has already been done for FIFO queues. The M/G/1 model might be directly applicable to communication networks if the service times are for bursts of packets (or cells) instead of individual packets, but we would not push that view. Instead, we contend that the M/G/1 asymptotic results are important primarily because they reveal important phenomena that should still be present in more complex models.

Let $W_2^c(t)$ be the ccdf of the low-priority steady-state waiting time (before beginning service). As in the FIFO case, there is often an exponential tail as in (1.1), but unlike the FIFO case, there also is routinely asymptotics of the form

$$W_2^c(t) \sim \alpha t^{-3/2} e^{-\eta t} \quad \text{as} \quad t \to \infty \tag{1.4}$$

for positive constants α and η. With FIFO we can also have asymptotics of the general form (1.4), but only for special service-time distributions. Unlike FIFO, asymptotics of the form (1.4) occurs for the low-priority waiting-time distribution for well-behaved service-time distributions, e.g., when both service-time distributions are exponential. Indeed, for well behaved service-time distributions, we have exponential asymptotics as in (1.1) when the low-priority arrival rate is above a critical threshold (with the requirement that the model still be stable), and the non-exponential asymptotics in (1.4) when the low-priority arrival rate is below this threshold. Moreover, when (1.4) holds, the asymptotic decay rate η in (1.4) does not increase (corresponding to smaller tail probabilities) when the low-priority arrival rate decreases. (This phenomenon can be understood by observing that the low-priority waiting time should be dominated by the remaining high-priority busy period when the low-priority arrival rate is sufficiently small.)

Our experience is that the non-exponential asymptotes, (1.4) and others such as (1.3), tend to be not nearly as accurate approximations as the exponential asymptote in (1.1), but the asymptotes indicate the general tendency. The quality of the different asymptotic approximations can be explained theoretically by the very different rates of convergence that prevail. The typical refinements of (1.1) and (1.4) are, respectively,

$$W_2^c(t) \sim \alpha e^{-\eta t}(1 + o(e^{-\epsilon t})) \quad \text{as} \quad t \to \infty \tag{1.5}$$

for $\epsilon > 0$ and

$$W_2^c(t) \sim \alpha t^{-3/2} e^{-\eta t}(1 + O(1/t)) \quad \text{as} \quad t \to \infty \,. \tag{1.6}$$

(Recall that an asymptotic expansion

$$f(t) \sim \sum_{k=0}^{\infty} g_k(t) \quad \text{as} \quad t \to \infty \tag{1.7}$$

means that

$$f(t) - \sum_{k=0}^{n} g_k(t) \sim g_{n+1}(t) \quad \text{as} \quad t \to \infty \,, \tag{1.8}$$

for all $n \geq 0$).

The exponential asymptotics with the exponential rate of convergence in (1.5) occurs when the rightmost singularity of the low-priority waiting-time Laplace transform $\hat{w}_2(s)$ is a simple pole at $-\eta$ with $Re(\phi) < -(\eta + \epsilon)$ for all other singularities ϕ. The non-exponential asymptotics in (1.6) occurs when the rightmost singularity of $\hat{w}_2(s)$ is a branch point singularity at $-\eta$, such that Heaviside's asymptotic expansion applies in the form

$$\hat{w}_2(s) \equiv \int_0^{\infty} e^{-st} dW_2(t) \sim \sum_{k=0}^{\infty} a_k (s + \eta)^{k/2} \quad \text{as} \quad s \to -\eta \,; \tag{1.9}$$

see p. 254 of Doetsch [10]. The whole integer powers in (1.9) correspond to analytic functions and thus have no influence upon the asymptotics as $t \to \infty$ in the time domain. We have established results of the form (1.5) and (1.6). We provide a conceptually satisfying derivation by obtaining a corresponding asymptotic expansion for the high-priority busy-period transform from Kendall's functional equation. The asymptotic expansion (1.9) for the transform $\hat{w}_2(s)$ follows quite directly from the asymptotic expansion for the busy-period transform by exploiting simple relations among asymptotic expansions.

2. The Geometric Random Sum Representation

In this section we express the classical results for the M/G/1 low-priority waiting-time transform in a convenient form. We consider two priorities, with high priority indexed by 1 and low priority indexed by 2. We consider the waiting time before beginning service in the non-preemptive case, which is the same as for preemptive resume; e.g., see (3.63) on p. 450 and (3.76) on p. 455 of Cohen [8]. Results for the waiting time to complete service – the overall sojourn time – can be obtained as corollaries.

For any *cumulative distribution function* (cdf) $G(t)$ on the nonnegative real line, let $g(t)$ be its *probability density function* (pdf), $G^c(t) = 1 - G(t)$ its *complementary cdf* (ccdf), g_k its k^{th} *moment*,

$$\hat{g}(s) = \int_0^{\infty} e^{-st} dG(t) = \int_0^{\infty} e^{-st} g(t) dt \tag{2.1}$$

its *Laplace-Stieltjes transform* (the Laplace transform of $g(t)$) and

$$\hat{G}_e(t) = \frac{1}{g_1} \int_0^t G^c(u) du \tag{2.2}$$

its associated *equilibrium excess cdf*, which has Laplace-Stieltjes transform $\hat{g}_e(s) = (1 - \hat{g}(s))/sg_1$.

For class i $(i = 1, 2)$, let the *arrival rate* be λ_i, the *service-time cdf* be $G_i(t)$ and the *traffic intensity* be $\rho_i \equiv \lambda_i g_{i1}$. To have proper steady-state distributions, we assume that

$$\rho \equiv \rho_1 + \rho_2 < 1 \ . \tag{2.3}$$

Without loss of generality (by choosing units), let $g_{11} = 1$.

The distribution of the low-priority waiting time depends on the busy-period distribution of the high priority class and related quantities. Let $B_1(t)$ be the class-1 busy-period cdf, $b_1(t)$ its density and $\hat{b}_1(s)$ the Laplace transform of $b_1(t)$, defined as in (2.1). The busy-period transform is characterized as the solution to the *Kendall functional equation*

$$\hat{b}_1(s) = \hat{g}_1(s + \rho_1 - \rho_1 \hat{b}_1(s)) \ . \tag{2.4}$$

Let

$$H_0^{(1)}(t) = (1 - P_{00}^{(1)}(t))/\rho \tag{2.5}$$

be the high-priority *server-occupancy cdf*, where $P_{00}^{(1)}(t)$ is the high-priority *emptiness probability*, i.e., the probability that the system has no class-1 customers at time t given that it had none at time 0, which is known to be monotone, so that $H_0^{(1)}(t)$ in (2.5) is a bonafide cdf; see Section 3 of [4]. Let $\hat{h}_0^{(1)}(s)$ be the Laplace-Stieltjes transform of $H_0^{(1)}(t)$, again defined as in (2.1). From Theorem 4 of [4],

$$\hat{h}_0^{(1)}(s) = \frac{\hat{b}_{1e}(s)}{1 - \rho_1 + \rho_1 \hat{b}_{1e}(s)} = \frac{1 - \hat{b}_1(s)}{s + \rho_1 - \rho_1 \hat{b}_1(s)} \ . \tag{2.6}$$

Let $F_{x0}^{(1)}(t)$ be the cdf of the *first passage time* to 0 for class 1 starting from an initial level of class-1 work x. The Laplace transform of its density $f_{x0}^{(1)}(t)$ is

$$\hat{f}_{x0}^{(1)}(s) = e^{-x z_1(s)} \ , \tag{2.7}$$

where

$$z_1(s) = s + \rho_1 - \rho_1 \hat{b}_1(s) \ ; \tag{2.8}$$

see (33) of [4]. The first passage time starting from a random level with density $g(t)$ having Laplace transform $\hat{g}(s)$ thus has Laplace transform $\hat{g}(z_1(s))$. The busy-period transform can thus be expressed as $\hat{b}_1(s) = \hat{g}_1(z_1(s))$ as in (2.4).

We now give a convenient expressions for the low-priority steady-state waiting-time transform.

Theorem 2.1. *The steady-state waiting time until beginning service for the low-priority class has Laplace transform*

$$\hat{w}_2(s) \equiv \int_0^\infty e^{-st} dW_2(t) = \frac{1 - \rho}{1 - \rho \hat{f}(s)} \ , \tag{2.9}$$

where

$$\hat{f}(s) = \frac{\rho_1}{\rho_1 + \rho_2} \hat{h}_0^{(1)}(s) + \frac{\rho_2}{\rho_1 + \rho_2} \hat{g}_{2e}(z_1(s)) \tag{2.10}$$

for $\hat{h}_0^{(1)}(s)$ *in (2.6),* $z_1(s)$ *in (2.8) and* $\hat{g}_{2e}(s) = (1 - \hat{g}_2(s))/g_{21}s$.

Formula (2.9) seems to be new, but it is elementary to show that it is equivalent to the transform formula in [8]. The representation (2.9) is attractive because it is in the same geometric-random sum form as the Pollaczek-Khintchine transform for the FIFO

waiting time with $\hat{f}(s)$ in (2.9) and (2.10) substituted for the transform of the equilibrium excess of the service-time distribution. The transform $\hat{f}(s)$ in (2.10), being a mixture of the two probability transforms $\hat{h}_0^{(1)}(s)$ and $\hat{g}_{2e}(z_1(s))$, is itself a transform of a probability distribution. Let $F(t)$ be the cdf associated with $\hat{f}(s)$ in (2.10), i.e.,

$$F(t) = \frac{\rho_1}{\rho_1 + \rho_2} H_0^{(1)}(t) + \frac{\rho_2}{\rho_1 + \rho_2} \int_0^\infty F_{x0}^{(1)}(t) dG_{2e}(x) \ . \tag{2.11}$$

Paralleling the classic Beneš formula for the M/G/1 FIFO waiting time, e.g., (4.82) on p. 255 of [8], we obtain a time-domain formula directly from (2.9). It is a geometric mixture of convolutions of the cdf F in (2.11).

Corollary. *The ccdf of the low-priority waiting time is*

$$W_2^c(t) = \sum_{n=1}^{\infty} (1-\rho)\rho^n F_n^c(t) \tag{2.12}$$

where $F_n^c(t)$ is the ccdf of the n-fold convolution of $F(t)$ in (2.11).

The moments of the low-priority waiting time are easily computed from (2.9), (2.11) and the moments of $H_0^{(1)}(t)$ and $F_{x0}^{(1)}(t)$, given in Theorems 6 and 7 of [4]. Unfortunately, the explicit formula for the ccdf $W_2^c(t)$ in (2.12) is not so convenient for calculation. However, it is straightforward to calculate the ccdf $W_2^c(t)$ numerically by inverting its transform

$$\hat{W}_2^c(s) = \int_0^\infty e^{-st} W_2^c(t) dt = \frac{1 - \hat{w}_2(s)}{s} \tag{2.13}$$

for $\hat{w}_2(s)$ in (2.9) [2]. We can obtain the required busy-period transform values $\hat{b}_1(s)$ for complex s by iterating the Kendall equation (2.4) [3]. In the longer version of this paper we have used the numerical transform inversion to calculate exact tail probabilities in order to evaluate the asymptotic approximations.

In closing this section, we remark that we have obtained convenient explicit results for the special cases in which (1) the high-priority service time is exponential and (2) the two service-time distributions coincide. For example, in the latter case $\hat{f}(s) = \hat{h}_0^{(1)}(s)$ for $\hat{f}(s)$ in (2.10) and $\hat{h}_0^{(1)}(s)$ in (2.6).

3. The Main Theorem

We how establish asymptotics for the low-priority waiting-time ccdf $W_2^c(t)$. Because of Theorem 2.1, we can draw on previous theory in the time domain, just as Pakes [15] did to treat non-exponential asymptotics for the GI/G/1 FIFO model. It turns out the asymptotics is determined by the *low-priority waiting-time root equation*

$$\hat{f}(s) = 1/\rho \ ; \tag{3.1}$$

a root of (3.1) is a zero of the denominator of the low-priority waiting-time transform $\hat{w}_2(s)$ in (2.9). Clearly, any root of equation (3.1) is a singularity of $\hat{w}_2(s)$. Moreover, since $\hat{f}(s)$ is monotone in real s where it is analytic, equation (3.1) has at most one root to the right of the rightmost singularity of $\hat{f}(s)$.

Theorem 3.1. *Let $-s^*$ be the rightmost singularity of $\hat{f}(s)$ in (2.10).*

(a) If equation (3.1) has a negative real root $-\eta$ with $\eta < s^*$, then

$$W_2^c(t) \sim \alpha e^{-\eta t} \quad \text{as} \quad t \to \infty \ , \tag{3.2}$$

where

$$\alpha = \frac{1-\rho}{-\rho\eta\hat{f}'(-\eta)} \ . \tag{3.3}$$

(b). If $\hat{f}(-s^*) < \rho^{-1}$, so that (3.1) has no root in $(-s^*, \infty)$, and other regularity conditions hold, then

$$W_2^c(t) \sim \frac{\rho(1-\rho)}{(1-\rho\hat{f}(-s^*))^2} F^c(t) \quad \text{as} \quad t \to \infty \tag{3.4}$$

for F in (2.11).

Further analysis is devoted to determining when (3.1) has a nonnegative zero and, when it does not, determining the asymptotic behavior of $F^c(t)$ in (3.4). Part (b) includes the long-tail case, for which we have obtained some results (not described here).

4. The Busy-Period Asymptotics Equation

Further detail about the asymptotics for $W_2^c(t)$ in Theorem 5.1 primarily depends upon asymptotics for the high-priority first-passage-time and busy-period distributions. We draw upon asymptotic results established by saddle point methods by Cox and Smith [9]. We extend their result to an asymptotic expansion. The reciprocal of the asymptotic decay rate for the busy period is the *relaxation time* for the total workload process; see III.7.3 of [8]. We let τ_1 represent this relaxation time.

Theorem 4.1. *If there is a negative real root $-\zeta_1$ of the busy-period asymptotics equation*

$$-\hat{g}_1'(s) = 1/\rho_1 \ , \tag{4.1}$$

where $-\zeta_1$ is to the right of all singularities of $\hat{g}_1(s)$, then $b_1(t)$ has the asymptotic expansion

$$b_1(t) \sim \alpha_1(\pi t^3)^{-1/2} e^{-t/\tau_1}(1 + \sum_{i=1}^{\infty} \beta_i t^{-i}) \quad \text{as} \quad t \to \infty \ , \tag{4.2}$$

where

$$\tau_1^{-1} = \rho_1 + \zeta_1 - \rho_1\hat{g}_1(-\zeta_1) \ , \quad \alpha_1 = [2\rho_1^3\hat{g}_1''(-\zeta_1)]^{-1/2} \tag{4.3}$$

and β_i is a constant for each i with

$$\beta_1 = -\alpha_1^4\rho_1^5\left[\frac{5}{3}\alpha_1^2\rho_1^3\hat{g}_1^{(3)}(-\zeta_1)^2 - \frac{\hat{g}_1^{(4)}(-\zeta_1)}{2}\right] \ . \tag{4.4}$$

We now characterize the rightmost singularity of $\hat{f}(s)$ under the assumption of Theorem 4.1 and determine when the exponential asymptotics holds. In our statement we use the inverse function of $z_1(s)$ for negative real s, denoted by $z_1^{-1}(s)$, which can be expressed as

$$z_1^{-1}(s) = s - \rho_1 + \rho_1\hat{g}_1(s) \ . \tag{4.5}$$

By (2.4), $z_1^{-1}(z_1(s)) = s$ for any complex s; see (29) and (30) of [4].

Theorem 4.2. *Assume the condition of Theorem 4.1.*

(a). *Let $-s_2^*$ and $-s^*$ be the rightmost singularities of $\hat{g}_2(s)$ and $\hat{f}(s)$. Then*

$$s^* = \begin{cases} \tau_1^{-1} \ , & \zeta_1 < s_2^* \\ -z_1^{-1}(-s_2^*) > 0 \ , & \zeta_1 > s_2^* > 0 \\ 0, & s_2^* = 0 \end{cases} \tag{4.6}$$

(b). *If in addition* $s_2^* > 0$, *then the root equation (3.1) has a negative real zero* $-\eta$ *with* $\eta < s^*$, *so that the exponential asymptotics in (3.2) holds, for all* $\rho > \rho_2^*$, *where*

$$\rho_2^* = \begin{cases} \frac{g_{21}\tau_1^{-1}}{\hat{g}_2(-\zeta_1)-1}\ , & s_2^* > \zeta_1 \\ 0\ , & s_2^* < \zeta_1 \text{ and } \hat{g}_2(-s_2^*) = \infty \\ \frac{1-\rho_1\hat{h}_0^{(1)}(z_1^{-1}(-s_2^*))}{\hat{g}_{2e}(-s_2^*)}\ , & s_2^* < \zeta_1 \text{ and } \hat{g}_2(-s_2^*) < \infty\ . \end{cases} \tag{4.7}$$

We now establish the non-exponential asymptotics in Theorem 3.1(b) in the main case.

Theorem 4.3. *If the conditions of Theorem 4.1 hold and if* $s_2^* > \zeta_1$, *where* $-s_2^*$ *is the rightmost singularity of the low-priority service-time transform* $\hat{g}_2(s)$, *then*

$$F^c(t) \sim \left(\frac{\rho_1\zeta_1^{-2} - \rho_2\rho_1\tau_1\hat{g}'_{2e}(-\zeta_1)}{\rho_1+\rho_2}\right)\alpha_1(\pi t^3)^{-1/2}e^{-t/\tau_1} \quad \text{as } t\to\infty \tag{4.8}$$

and, if $\rho_2 < \rho_2^*$ *for* $\rho_2^* = g_{21}/\tau_1(\hat{g}_2(-\zeta_1)-1)$ *as in (4.7), then*

$$W_2^c(t) \sim \frac{(1-\rho)\rho_1\tau_1^2[1-\rho_2\tau_1\zeta_1\hat{g}'_2(-\zeta_1)g_{21}^{-1} - (\rho_2/\rho_2^*)]}{[1-(\rho_2/\rho_2^*)]^2}\ \frac{\alpha_1 e^{-t/\tau_1}}{\sqrt{\pi t^3}} \quad \text{as } t\to\infty \tag{4.9}$$

for ζ_1 *in (4.1),* τ_1^{-1} *in (4.3) and* α_1 *in (4.3).*

Note that the asymptotic decay rate τ_1^{-1} in (4.9) does not depend on ρ_2 provided that $\rho_2 < \rho_2^*$. Note that the asymptotic constant in (4.9) explodes as $\rho_2 \uparrow \rho_2^*$.

5. Asymptotic Expansions from Kendall's Equation

In Theorem 4.1 we obtained an asymptotic expansion for the high-priority busy-period pdf $b_1(t)$ by using saddle point methods with an integral representation. This enabled us to obtain the asymptote for the low-priority waiting-time ccdf $W_2^c(t)$, but this required rather difficult arguments in the time domain. Asymptotics for $W_2^c(t)$ can be developed more easily starting from an asymptotic expansion for the high-priority busy-period transform $\hat{b}_1(s)$.

We obtain our asymptotic expansion for $\hat{b}_1(s)$ from Kendall's functional equation (2.4). If we assume that the rightmost singularity of $\hat{b}_1(s)$ is a branch point singularity at $s = -\tau_1^{-1}$ and that $b_1(s)$ no other singularities for $Re(s) \geq -\tau_1^{-1} - \epsilon$ for some positive ϵ, and if we also assume that $\hat{b}_1(s)$ has an asymptotic expansion in powers of $(s+\tau_1^{-1})$, i.e.,

$$\hat{b}_1(s) \sim \sum_{k=0}^{\infty} a_k(s+\tau_1^{-1})^{\gamma_k} \quad \text{as } s\to -\tau_1^{-1}\ , \tag{5.1}$$

where $\gamma_k < \gamma_{k+1}$ for all k, then we can apply Theorem 4.1 and Heaviside's theorem on p. 254 of [10] to conclude that $\hat{b}_1(s)$ must have an asymptotic expansion in half powers, i.e.,

$$\hat{b}_1(s) \sim \sum_{k=0}^{\infty} a_k(s+\tau_1^{-1})^{k/2} \quad \text{as } s\to -\tau_1^{-1}\ , \tag{5.2}$$

when the busy-period root equation (4.1) has a root to the right of all singularities to the high-priority service-time transform $\hat{g}_1(s)$. The terms in (5.1) with non-integer powers must have counterparts in the time-domain expansion in Theorem 4.1. As indicated in [10], the terms $a_{2k}(s+\tau_1^{-1})^k$ in (5.1) and (5.2) with whole integer powers are analytic

functions and thus do not contribute to the asymptotic expansion in the time domain. Since the terms with integer powers in (5.2) have no counterpart in the time domain, evidently it is not possible to obtain the transform asymptotic expansion (5.2) directly from the time-domain asymptotic expansion established in Theorem 4.1, but that is an unresolved point.

We now directly establish the asymptotic expansion in half powers in (5.2). Thus, we obtain an asymptotic expansion for $\hat{b}_1(s)$ and an alternative proof of Theorem 4.1. To carry out one step of the proof of Theorem 5.1 below and to understand how Theorems 4.1 and 5.1 are related, we need to know how Heaviside's theorem converts asymptotics for transforms into asymptotics in the time domain. Let $\mathcal{L}^{-1}(\hat{f}(s))$ denote the inverse of the transform $\hat{f}(s)$. Then

$$\mathcal{L}^{-1}(\sqrt{s+\tau_1^{-1}}) \sim \frac{-e^{-t/\tau_1}}{2\sqrt{\pi t^3}} \quad \text{and} \quad \mathcal{L}^{-1}((s+\tau_1^{-1})^{3/2}) \sim \frac{3}{4}\frac{e^{-t/\tau_1}}{\sqrt{\pi t^5}} \quad \text{as} \quad t \to \infty \; ; \tag{5.3}$$

see p. 254 of [10]. In particular, note the sign change in the first case.

Theorem 5.1. *If the busy-period equation in (4.1) has a root $-\zeta_1$ with $-s_1^* < -\zeta_1$, where $-s_1^*$ is the rightmost singularity of $\hat{g}_1(s)$, then the half-power asymptotic expansion (5.2) is valid and the first four coefficients in (5.2) are*

$$a_0 = \hat{g}_1(-\zeta_1) \; , \quad a_1 = \frac{-2}{\sqrt{2\rho_1^3 \hat{g}_1^{(2)}(-\zeta_1)}} = -2\alpha_1 \; , \tag{5.4}$$

$$a_2 = \frac{1}{\rho_1} + \frac{(a_1\rho_1)^4}{12}\hat{g}_1^{(3)}(-\zeta_1) \; , \tag{5.5}$$

and

$$a_3 = \frac{-(a_1\rho_1)^5}{48}\left[\hat{g}_1^{(4)}(-\zeta_1) - \frac{5}{6}a_1^2\rho_1^3\hat{g}_1^{(3)}(-\zeta_1)^2\right] \tag{5.6}$$

for α_1 in (4.3).

Using (5.3), we can see that the coefficients a_1 and a_3 in (5.4) and (5.6) are consistent with the two calculated coefficients in Theorem 4.1. In Theorem 4.1 there is no counterpart of a_2 in (5.5) though.

We now can apply the asymptotic expansion for $\hat{b}_1(s)$ in (5.2) to obtain asymptotic expansions for the other quantities of interest by using elementary operations on asymptotic expansions. We successively obtain asymptotic expansions for $\hat{h}_0^{(1)}(s)$, $\hat{g}_{2e}(z_1(s))$ and $1-\rho\hat{f}(s)$ to obtain the following result. There are three cases, depending on whether ρ_2 is greater than, equal to or less than ρ_2^* for ρ_2^* in (4.7). Let e_0 and e_1 be defined by

$$e_0 = [1-(\rho_2/\rho_2^*)]/\zeta_1\tau_1 \; , \quad e_1 = 2\alpha_1\rho_1(1-\rho_2\tau_1\zeta_1^2\hat{g}'_{2e}(-\zeta_1))/\tau_1\zeta_1^2 \tag{5.7}$$

Theorem 5.2. *Assume that the busy-period equation (4.1) has a root $-\zeta_1$ with $-s_1^* < -\zeta_1$ and $-s_2^* < -\zeta_1$, where $-s_i^*$ is the rightmost singularity of $\hat{g}_i(s)$. Assume that $\hat{b}_1(s)$ has no singularities besides $-\tau_1^{-1}$ with $Re(s) > -\tau_1^{-1} - \epsilon$. Let η, α, $\alpha_1 = -a_1/2$, e_0, e_1 be as in (3.1), (3.3), and (4.3) and (5.7).*

(a) If $\rho_2 > \rho_2^*$ for ρ_2^* in (4.7), then

$$\hat{w}_2(s) - \frac{\alpha\eta}{s+\eta} \sim \sum_{k=0}^{\infty} v_k(s+\tau_1^{-1})^{k/2} \quad \text{as} \quad s \to -\tau_1^{-1} \tag{5.8}$$

for constants v_k and, in particular,

$$v_0 = \frac{1-\rho}{e_0} \quad \textit{and} \quad v_1 = \frac{-(1-\rho)e_1}{e_0^2}\,, \tag{5.9}$$

so that

$$w_2(t) - \alpha\eta e^{-\eta t} \sim \frac{A_1\alpha_1 e^{-t/\tau_1}}{\sqrt{\pi t^3}}(1+O(1/t)) \quad \text{as} \quad t\to\infty \tag{5.10}$$

and

$$W_2^c(t) - \alpha e^{-\eta t} \sim \frac{A_1\alpha_1\tau_1 e^{-t/\tau_1}}{\sqrt{\pi t^3}}(1+O(1/t)) \quad \textit{as} \quad t\to\infty\,, \tag{5.11}$$

where

$$A_1 = \frac{(1-\rho)e_1}{2\alpha_1 e_0^2} = \frac{(1-\rho)\rho_1\tau_1(1-\rho_2\tau_1\zeta_1^2\hat{g}'_{2e}(-\zeta_1))}{[1-(\rho_2/\rho_2^*)]^2}\,. \tag{5.12}$$

(b) If $\rho_2 = \rho_2^*$, then $e_0 = 0$ and

$$\hat{w}_2(s) \sim \sum_{k=-1}^{\infty} v_k(s+\tau_1^{-1})^{k/2} \quad \textit{as} \quad s\to -\tau_1^{-1} \tag{5.13}$$

and, in particular,

$$\hat{w}_2(s) \sim \frac{1-\rho}{e_1\sqrt{s+\tau_1^{-1}}} \quad \textit{as} \quad s\to -\tau_1^{-1}\,, \tag{5.14}$$

so that

$$w_2(t) \sim A_2\frac{e^{-t/\tau_1}}{\sqrt{\pi t}}(1+O(1/t)) \quad \textit{as} \quad t\to\infty \tag{5.15}$$

and

$$W_2^c(t) \sim A_2\tau_1\frac{e^{-t/\tau_1}}{\sqrt{\pi t}}(1+O(1/t)) \quad \textit{as} \quad t\to\infty\,, \tag{5.16}$$

where

$$A_2 = \frac{1-\rho}{e_1} = \frac{(1-\rho)\tau_1\zeta_1^2}{2\alpha_1\rho_1(1-\rho_2^*\tau_1\zeta_1^2\hat{g}'_{2e}(-\zeta_1))}\,. \tag{5.17}$$

(c) If $\rho_2 < \rho_2^*$, then

$$\hat{w}_2(s) \sim \sum_{k=0}^{\infty} v_k(s+\tau_1^{-1})^{k/2} \quad \textit{as} \quad s\to -\tau_1^{-1}\,, \tag{5.18}$$

where v_0 and v_1 are given in (5.9), so that

$$w_2(t) \sim \frac{A_1\alpha_1 e^{-t/\tau_1}}{\sqrt{\pi t^3}}(1+O(1/t)) \quad \textit{as} \quad t\to\infty \tag{5.19}$$

and

$$W_2^c(t) \sim A_1\alpha_1\frac{\tau_1 e^{-t/\tau_1}}{\sqrt{\pi t^3}}(1+O(1/t)) \quad \textit{as} \quad t\to\infty\,, \tag{5.20}$$

where A_1 is in (5.12).

The constant A_1 in (5.12) is consistent with the asymptotic constant for the ccdf $W_2^c(t)$ in (4.9).

By Theorem 5.2, $\alpha\to 0$ as $\rho_2\downarrow\rho_2^*$ where α is the first asymptotic constant in (5.10). On the other hand, by (5.12), $e_0\to 0$ and $A_1\to\infty$ as $\rho_2\downarrow\rho_2^*$. Hence, even the two-term approximation based on (5.10) cannot be good when ρ_2 is close to ρ_2^*.

References

[1] J. Abate, G. L. Choudhury and W. Whitt, Asymptotics for steady-state tail probabilities in structured Markov queueing models, *Stochastic Models* 10 (1994) 99–143.

[2] J. Abate and W. Whitt, The Fourier-series method for inverting transforms of probability distributions, *Queueing Systems* 10 (1992) 5–88.

[3] J. Abate and W. Whitt, Solving probability transform functional equations for numerical inversion, *Opns. Res. Letters* 12 (1992) 275–281.

[4] J. Abate and W. Whitt, Transient behavior of the M/G/1 workload process, *Opns. Res.* 42 (1994) 750–764.

[5] C. S. Chang and J. A. Thomas, Effective bandwidths in high-speed digital networks. *IEEE J. Sel. Areas Commun.* 13 (1995) 1091–1100.

[6] G. L. Choudhury, D. M. Lucantoni and W. Whitt, Squeezing the most out of ATM, *IEEE Trans. Commun.* 44 (1996) 203–217.

[7] G. L. Choudhury and W. Whitt, Long-tail buffer content distributions in broadband networks, *Perf. Eval.* 26 (1997) to appear.

[8] J. W. Cohen, *The Single Server Queue*, Revised edition, North-Holland, Amsterdam, 1982.

[9] D. R. Cox and W. L. Smith, *Queues*, Methuen, London, 1961.

[10] G. Doetsch, *Introduction to the Theory and Applications of the Laplace Transformation*, Springer, New York, 1974.

[11] N. G. Duffield and N. O'Connell, large deviations and overflow probabilities for the general single-server queue, with applications, *Math. Proc. Camb. Phil. Soc.*, 118 (1995) 363–374.

[12] P. W. Glynn and W. Whitt, Logarithmic asymptotics for steady-state tail probabilities in a single-server queue, *Studies in Applied Probability, Papers in Honor of Lajos Takács*, J. Galambos and J. Gani, eds., Applied Probability Trust, Sheffield, U.K., 1994, 131–146.

[13] P. R. Jelenković and A. A. Lazar, Subexponential asymptotics of a Markov-modulated G/G/1 queue, Department of Electrical Engineering, Columbia University, 1995.

[14] W. E. Leland, M. S. Taqqu, W. Willinger and D. V. Wilson, On the self-similar nature of ethernet traffic, *ACM SIGCOMM'93*, (1993), 183–193.

[15] A. G. Pakes, On the tails of waiting-time distributions, *J. Appl. Prob.* 12 (1975) 555–564.

INTRODUCTION TO FUZZY SYSTEMS : FROM FUZZY SETS TO INDUSTRIAL APPLICATIONS

MOHAMMAD JAMSHIDI
AT & T Professor and Director
NASA Center for Autonomous Control Engineering
University of New Mexico
Albuquerque, NM 87131 USA
Telephone : 505-277-5538
FAX : 505-277-4681
E-mail : jamshidi@unm.edu
WWW: http://ace.unm.edu

Abstract - One of the more popular new technologies of today is "intelligent control," which is defined as a combination of control theory, operations research, and artificial intelligence (AI). Among many possible new technologies based on AI, fuzzy logic is now perhaps the most popular area, judging by the billions of dollars worth of sales of products and close to 2000 patents issued in Japan alone since the announcement of the first fuzzy chips in 1987. This picture is not however so booming in the USA. In recent years from 700 fuzzy logic patents issued by the US Patent office, only 60 of them were issued to non-Japanese inventors. Thanks to tremendous technological and commercial advances in fuzzy logic in Japan and other nations, today fuzzy logic is enjoying an unprecedented popularity in the technological and engineering fields including manufacturing, automation, financial engineering (e.g. stock markets, mutual funds, etc.), medicine, etc. The most important issue facing many industrialized nations in the next several decades will be global competition to an extent that has never before been posed. The purpose of this paper is to introduce fuzzy logic and fuzzy control to readers . It describes its birth and progress up to now.

1. INTRODUCTION

In 1989, in a study performed for the US Congress, the United States Office of Technology Assessment studied more than 12 competing technologies to reduce costs in space applications. The number one technology on their list turned out to be "expert systems," including "fuzzy expert systems." Fuzzy logic and fuzzy expert control systems have found applications in numerous appliances and systems with moderate size and limited number of components. One of the first complex systems in which fuzzy logic has been successfully applied is cement kilns, which began in 1977 in Denmark. Today, most of the world's cement kilns are using some type of fuzzy expert system. The bulk of today's applications are

undertaken by the Japanese industries. In broad sense, fuzzy logic refers to fuzzy sets which was introduced in the USA by Zadeh [1]. A fuzzy set has movable boundaries, i.e., the elements of such sets not only represent the colors black and white, but also allow a spectrum of gray colors in between.

Currently, one of the more active areas of fuzzy logic applications is control systems. Fuzzy controllers are expert control systems that smoothly interpolate between hard-boundary crisp rules. Rules fire simultaneously to continuous degrees or strengths and the multiple resultant actions are combined into an interpolated result. Processing of uncertain information and saving of energy using common sense rules and natural language statements are the basis for fuzzy control. The use of sensor data in practical control systems involves several tasks that are usually done by a human in the decision loop, e.g., an astronaut adjusting the position of a satellite or putting it in the proper orbit, a driver adjusting a vehicle's air-conditioning unit, etc. All such tasks must be performed based on the evaluation of data according to a set of rules in which the human expert has learned from experience or training. Often, if not all the time, these rules are not crisp, i.e., some decisions are based on common sense or personal judgment. Such problems can be addressed by a set of fuzzy variables and rules which, if properly constructed, can make decisions as well as an expert.

This paper represents the introduction to fuzzy logic and its most important application - *fuzzy control*. The structure of the paper is as follows: Section 2 gives a brief introduction into fuzzy logic. An introduction to fuzzy control appears in Section 3. Section 4 constitutes some applications of fuzzy control, while discussions and conclusions are described in Section 5.

2. FUZZY LOGIC

The need and use of multi-level logic can be traced from the ancient works of Aristotle, who is quoted as saying, "There will be a sea battle tomorrow." Such a statement is not yet true or false, but is potentially either. Much later, around AD 1285-1340, William of Occam supported two-valued logic, but speculated on what the true value of "if p then q" might be if one of the two components were neither true nor false. During the time period of 1878-1956, Lukasiewicz proposed a three-level logic as a "true" (1), a "false" (0), and a "neuter" (1/2), which represented half true or half false. In subsequent times, logicians in China and other parts of the world continued on the notion of multi-level logic and proposed multi-level logic. Zadeh, in his 1965 seminal paper [1], finished the task by following through the speculation of previous logicians and showing that what he called "fuzzy sets" were the foundation of *any* logic, regardless of the number of truth levels. He chose the word "fuzzy" for the continuum of logical

values between 0 (completely false) and 1 (completely true). The theory of fuzzy logic deals with two problems of: i) the fuzzy set theory, which deals with the ambiguity found in semantics, and ii) the fuzzy measurement theory, which deals with the ambiguous nature of judgments and evaluations.

The primary motivation and "banner" of fuzzy logic is the possibility of exploiting tolerance for some inexactness and imprecision. Precision is often very costly, so if a problem does not require high precision, one should not have to pay for it. The traditional example of parking a car is a noteworthy illustration. If the driver is not required to park the car within an exact distance from the curb, why spend any more time than necessary on the task as long as it is a legal parking operation. Fuzzy logic and classical logic differ in the sense that the former can handle both symbolic and numerical manipulation, while the latter can handle symbolic manipulation only. In a broad sense, fuzzy logic is a union of fuzzy (fuzzified) crisp logics [2]. To quote Zadeh, "Fuzzy logic's primary aim is to provide a formal, computationally-oriented system of concepts and techniques for dealing with modes of reasoning which are approximate rather than exact." Thus, in fuzzy logic, exact (crisp) reasoning is considered to be the limiting case of approximate reasoning. In fuzzy logic one can see that everything is a matter of degree.

In an attempt to translate the crisp knowledge in a process, such as voltage across a terminal, to a linguistic or fuzzy knowledge -- that is, to go through the process of "*fuzzification* " -- one must make the binary input and output variables of a plant members of some fuzzy set, e.g. the set of bright images on the focal plane of a telescope or the set of small voltages across the armature of a DC motor. Fuzzy sets may be represented by a mathematical formulation often known as the membership function. This function gives a degree or grade of membership within the set. The membership function of a fuzzy set A , denoted by $\mu A(X)$, maps the elements of the universe X into a numerical value within the range [0,1], i.e.,

$$\mu A(X) : X \text{ ----------> } [0,1].$$

Where $x \varepsilon X$. Note that a membership function is a so-called possibility function and not a probability function (see Jamshidi, et al. [2]). Typical membership functions for various fuzzy (linguistic) variables (sets) are triangular, trapezoidal, Gaussian, or a singleton functions. Within this framework, a membership value of zero corresponds to a value which is definitely not an element of the fuzzy set, while a value of 1 corresponds to the case where the element is definitely a member of the set. In fuzzy logic, like binary logic, operations such as union, intersection, complement, OR, AND, etc., are all defined.

3. FUZZY CONTROL

Fuzzy control systems are rule-based systems in which a set of so-called fuzzy rules represents a control decision mechanism to adjust the effects of certain causes coming from the system. The aim of fuzzy control systems is normally to substitute for or replace a skilled human operator with a fuzzy rule-based system. Figure 1 shows a block diagram for this definition. As shown, the human operator observes quantities by reading a meter or assessing a chart (i.e., noting fuzzy variables) and then performing a definite action, such as pushing a knob or turning a wheel (i.e., providing a crisp action or output y (see Figure 1). In a similar fashion the fuzzy controller, shown in Figure 1, uses crisp data directly from a number of sensors; through the process of fuzzification these are changed to linguistic or fuzzy membership functions (fuzzified). They then go through a set of fuzzy "IF-THEN" rules in an inference engine much like an expert system and result in some fuzzy output(s) designated by variable z. The fuzzy output(s) will then be changed back into crisp values through a process called *"defuzzification"* by some weighted average method such as the "centroidal" method . The result is a value of the output y*. In this way, the designer has obtained an approximate output y* for the actual output y.

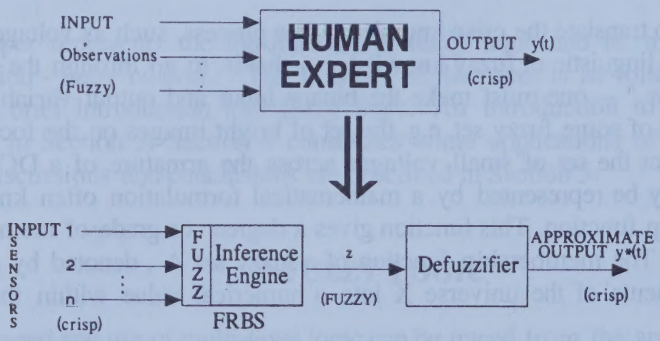


Figure 1. A conceptual definition of a fuzzy control system.

As indicated before, a fuzzy controller typically takes the form of a set of IF-THEN rules whose *antecedents* (IF part) and *consequents* (THEN part) are themselves membership functions. Consequents from different rules are numerically combined (typically union via MAX) and are then collapsed (typically taking the centroid of the combined distribution) to yield a single real-number (binary) output. Within the framework of a fuzzy expert system, like regular expert systems, typical rules can be the result of a human operator's knowledge, e.g.:

"IF the Temperature is Hot
THEN decrease the Current to a Medium level."

In this rule, *Hot* and *Medium* are fuzzy variables. Such natural language rules can then be translated into typical computer language statements to be coded in a software environment.

Using these rules, an entire finite number of rules can be derived in the form of natural language statements as if a human operator were performing the controlling task. In any practical system, such as an air conditioning system, the user or an operator often fine-tunes, tweaks, and adjusts the knobs until the desired cool (or hot) air can be felt with the desired speed. Such operator knowledge can be utilized in the design of a fuzzy controller for an air conditioning unit system. One of the most common ways of designing a fuzzy controller is through "fuzzy rule-based systems." Figure 2 shows a typical laboratory implementation of a fuzzy control system. The controller shows the processes of fuzzification, (i.e., binary to fuzzy transformation) and defuzzification (i.e., fuzzy to binary transformation).

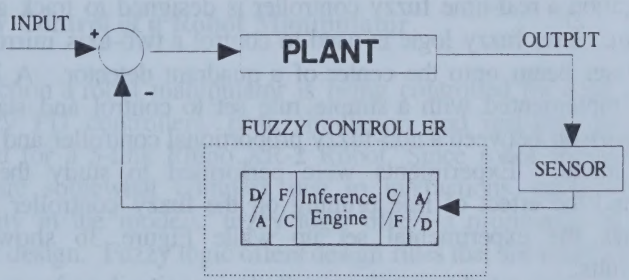


Figure 2. Block diagram for a laboratory implementation of a fuzzy controller.

An alternative way of implementing a fuzzy control regime, which is similar to a conventional adaptive control law, is to use a standard crisp logic controller such as, say a PID ($u = K_p.e(t) + K_i \int e(t)dt + K_d\, de(t)/dt$),
and then use fuzzy IF-THEN rules to "tune" the gains of the conventional controller, i.e., Kp Ki, and Kd. Here variable e(t) is the tracking error for a system output y(t) which is to follow a desired output yd(t).

The fuzzy control problem, like any control problem, is one of evaluating a nonlinear mapping h(.), defined by:

h : e ----------> u

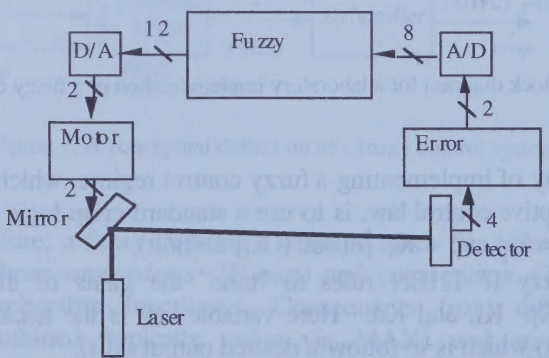
where u and e are the control and the error signals, respectively. The choice of h(.) is the essence of the control problem. For a two-sensor case, i.e., a two-error variable (say e and Δe) and one control signal, the plot of u versus e and Δe will provide a surface this book will call *control surface*. As indicated earlier, fuzzy controllers are expert control systems that smoothly interpolate between crisp rules. This often results in a savings of energy, because the fuzzy control surface fits underneath the expert control surface.

4. APPLICATIONS

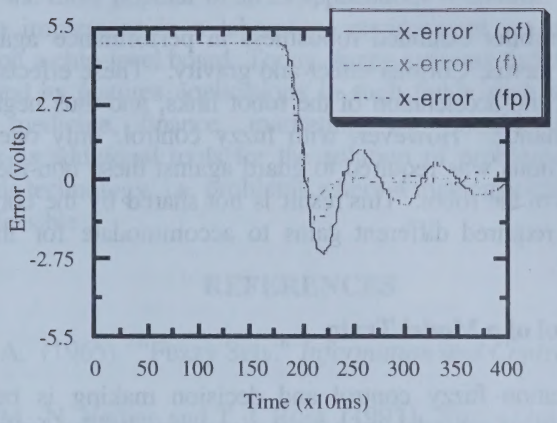
In this section a limited number of applications of fuzzy control systems stemming from the work done at CAD Laboratory for Intelligent and Robotics Systems and NASA Center for Autonomous Control Engineering (ACE) at the University of New Mexico. Some of these applications are documented in a book edited by the author and his colleagues [2] .

4.1 Fuzzy Control of a Laser Beam

In this application a real-time fuzzy controller is designed to track a Helium-Neon laser beam. Here, fuzzy logic is used to control a two-axis mirror gimbal for aligning a laser beam onto the center of a quadrant detector. A PC based controller was implemented with a simple rule set to control and stabilize the device. A comparison between a non-fuzzy proportional controller and the fuzzy controller was made. Experiments were performed to study the tracking characteristics and the effect of rule pruning on the fuzzy controller response. Figure 3a shows the experimental set up while Figure 3b shows typical experimental results.



(3a) System configuration.



3b) Typical Experimental Result

Figure 3 . Experimental set-up for a laser-tracking fuzzy control system

4.2 Fuzzy Control of a Robot Manipulator

In this section a robot manipulator is being controlled by a set of independent-joint fuzzy PD controllers. Both simulation and real-time experiments were performed for a 5-link Rhino XR-2 Robot. Since robot manipulators dynamic Models are somewhat complex due to interactions among links/joints and nonlinearity in the models, fuzzy logic offers a promising approach to robot controller design. Fuzzy logic offers design rules that are relatively easy to use in a wide range of applications, including nonlinear robotic equations.

Many test were made for the experimental phase of this application. For example, in one of tests, the controllers were compared under various ranges of operation. The PD controller with suitable gain improved trajectory response by reducing the overshoot to less than four counts. However, the PD controller was unable to move the motor when the links were initially only one or two counts away from the final destination. The primary cause for this effect was the nonlinear friction in the robot joints. One could overcome this resistance with an increase in controller gain, but then the existing trajectory overshoot would be intensified.

The fuzzy controller was able to overcome the problems encountered by traditional PD control and perform better in all conditions tested. Results showed that the maximum overshoot with fuzzy control was held to a count of one. The

fuzzy controller also was able to actuate the motor and travel a short distance of one count, unlike the PD controller.

The fuzzy controller exhibited robustness in performance against non-ideal effects like robot inertia, Coriolis effect and gravity. These effects influence the velocity, position, and acceleration of the robot links, and thus negatively impact controller performance. However, with fuzzy control, only one set of fuzzy membership functions was required to guard against these non-ideal effects and operating all links of the robot. This result is not shared by the conventional PD controller which required different gains to accommodate for the differences between each link.

4.3 Fuzzy Control of a Model Train

In this application fuzzy control and decision making is being used for acceleration control of a model train on a circular path. The fuzzy acceleration control system showed a marked improvement on smoothness of ride of the fuzzy control versus crisp control.

Previous software experiences with the applications of fuzzy control to traffic {see Jamshidi, et al., [2], Chapter 12), has shown very promising results. In over 625 different simulation runs, it was shown that nearly one half of the waiting time behind red lights in a single intersection can be saved as compared with standard techniques such as sensor-based or timed cycle approaches. A comparison of the cost (waiting time) was made for fuzzy controller and two standard methods. The fuzzy controller improved on the waiting time, thereby cost, somewhat drastically.

Here an attempt was made to use fuzzy logic to control the acceleration of a model train. Velocity is a property which humans do not notice. It is the acceleration that can cause discomfort in a passenger. By selecting a comfortable acceleration and maintaining it as a constant, a smooth increase in velocity and thus a smooth and comfortable ride can be achieved. The acceleration problem is one of maintaining a constant acceleration and/or deceleration for a moving object (for example a train) has a fuzzy nature. Fuzzy logic was chosen as a means of control for this problem. The idea being that a constant acceleration needs to be achieved as quickly as possible and then maintained. This is also true of deceleration when bringing the train to a stop

5. CONCLUSIONS

The basic theme of this paper was fuzzy logic and fuzzy control with applications. An attempt was made to introduce the technology and put an emphasize on the most popular of all its applications -- control. Fuzzy controllers are simple to implement in a laboratory environment on either a Personal Computer or on a chip-level board. Today, fuzzy systems technology is a matter of economy and its features applications in such fields such as control, pattern recognitions, medicine, finance, marketing, etc. should be given serious considerations as additional tools for the solution of problems which are most suitable for this technology, i.e. problems where a mathematical model is neither available nor feasible .

REFERENCES

1. Zadeh, L. A. (1965). "Fuzzy Sets." *Information and Control*, Vol. 8, pp. 335-353.
2. Jamshidi, M., N. Vadiee and T. J. Ross (1993). *Fuzzy Logic and Control - Software and Hardware Applications*. Prentice Hall Series on Environmental and Intelligent Manufacturing Systems (M. Jamshidi, Ed.), Vol. 3.

Concurrent FL-PI Controller Designed Using GAs

Guomin Li K. M. Tsang Hui Li
Dept. of Electrical Engineering, Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong

Abstract: A novel scheme for hybrid controller --- concurrent controller --- is proposed, in which a FL sub-controller is in shunt with a PI sub-controller and the overall control action is the sum of contribution of each sub-controller. The scaling factors of FL sub-controller and parameters of PI sub-controller are optimized using GAs. The proposed scheme is applied to the position control of a DC servomotor system. Simulation results demonstrate that the concurrent controller has advantages of two control methods but does not produce the switching chattering, and hence it is truly superior to both the single FL controller and the single PI controller.

1. Introduction

Since Mamdani et al first applied Fuzzy Logic (FL) to the steam engine control[1] based on the mathematical foundation of fuzzy system by Zadeh [2], Fuzzy Logic has found wide applications in different areas of industry [3], including in fields of power electronics [4] and motion control [5] [6] [7]. Fuzzy Logic is a kind of promising Artificial Intelligence (AI) techniques. It emulates the human thinking and human inference, and often yields satisfactory performances in applying to control systems. Moreover, Fuzzy Logic can capture the approximate and inexact natures of the real world and possesses some adaptability to the changing environments. Therefore, it also produces good robustness. Nevertheless, disadvantages of Fuzzy Logic based Controller (FLC) are also found in its applications to plants which have inherently fast response characteristics, such as motor servomechanisms. In those cases, FLC might not respond as quickly as some of other controllers, e.g. the conventional PI controllers. Furthermore, it might introduce steady-state errors to the controlled systems.

To overcome disadvantages of single control method and integrate advantages of different control methods into a same system, hybrid control has become one of the attractive research issues. Kim et al proposed a fuzzy precompensated PID controller for DC servomotor position control [8]. In their scheme, FLC is used to modify the command input of system and therefore it is essentially a compensator. Matsunaga et al presented a hybrid scheme also for controlling a DC servomotor [9], in which the controller switches between a fuzzy controller and a PID controller depending on whether the system is in transient progress or steady state. It is easily understood that the output of whole controller in such a scheme will display the discontinuity at the switching instant and hence the chattering is difficult to avoid.

Genetic Algorithms (GAs) are very powerful algorithms for searching globally optimal solutions to given problems. They emulate biological evolutionary theories rather than rely on computing local derivatives to guide the parallel search process.

They are important tools not only in the function optimization but also in the controller design [10] . Their applications to the design of FLCs are getting more and more active these years. Among the related works, however, most efforts were mainly devoted to the optimization of membership functions [11] and control rules [12] .

In this paper, a novel hybrid controller --- concurrent controller --- is proposed, in which a FL sub-controller is in shunt with a PI sub-controller and the overall control action is the sum of contribution of each sub-controller. The optimal scaling factors of FL sub-controller and optimal parameters of PI sub-controller are designed using GAs. The proposed scheme is applied to the position control of a DC servomotor system. Simulation results demonstrate that the concurrent controller has advantages of two control methods but does not produce the annoying switching chattering, and hence it is really superior to both the single FL controller and the single PI controller.

2. The Scheme of Concurrent Controller

FLC does not need precise dynamics of the controlled plant. It gives control action based on a set of control rules which reflect the expert's knowledge and/or the operator's experience. FLC also adapts itself to the load disturbances and model variations to certain extent. Therefore, it produces much better robustness compared with conventional controllers. However, its defects in lack of ability to respond fast enough and in possible production of steady-state error have to be compensated in some high performance control systems such as position tracking systems.

The conventional PI controller responds quickly and its integral control action has the potential to eliminate the steady-state error of system. It can be predicted that incorporating a FL controller with a PI controller to form a hybrid controller would yield remarkable results. However, in most of present hybrid schemes, usually only one sub-controller takes effect at any instant and the overall controller switches between sub-controllers. Hence, the chattering is difficult to avoid.

In this paper, a novel hybrid controller --- concurrent controller --- is proposed. Fig 1 shows its block diagram, in which a FL sub-controller is in parallel with a PI sub-controller. The synthetic control action is directly the sum of output of each sub-controller. It takes effect after passing through a saturation element that mimics the most common constraint imposed on the output of actual controller.

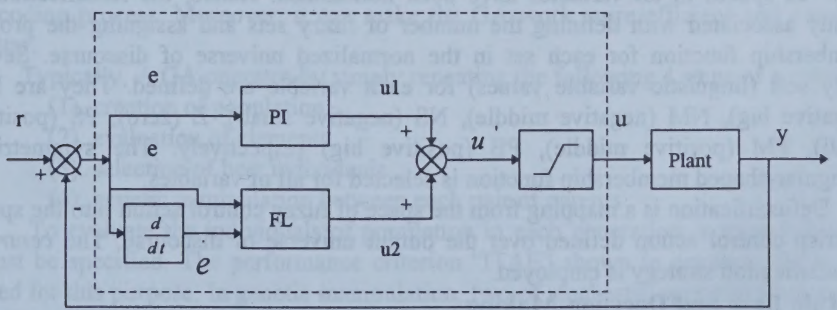


Fig 1 Scheme of Concurrent Controller

In the transient progress, two sub-controllers operate in parallel. In the steady state, the PI sub-controller plays the dominant control role. This is because the output of FL sub-controller becomes almost zero at this state. Hence, it could be considered that the controller has switched to the single PI controller at this time. However, this "switching" is naturally and smoothly completed. It essentially differs from the "hard switching" that exists in most of present hybrid control schemes. As a result, the chattering does not emerge.

3. GA based Design of FL Sub-Controller

FL sub-controller takes the same configuration as a general FL controller. Fig 2 gives its block diagram. It has two input variables and one output variable. By their own scale mapping blocks with nonlinear saturation, the actual ranges of error and change of error are all transformed into normalized universes of discourse, i.e., the closed intervals with the form [-1, 1], while the normalized output universe of discourse is converted to the actual output range by being multiplied by its scaling factor.

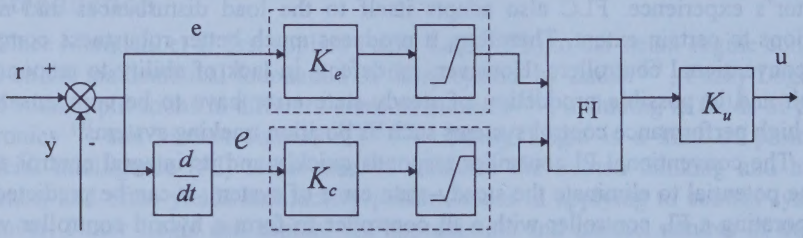


Fig 2 block diagram of FL sub-controller

1). Fuzzification and Defuzzification

In general, fuzzification could be defined as a mapping from the observed space of variable to fuzzy sets in its universe of discourse. In the proposed scheme, the observed spaces of all variables have been normalized. Hence, the fuzzification is mainly associated with defining the number of fuzzy sets and assigning the proper membership function for each set in the normalized universe of discourse. Seven fuzzy sets (linguistic variable values) for each variable are defined. They are NB (negative big), NM (negative middle), NS (negative small), Z (zero), PS (positive small), PM (positive middle), PB (positive big) respectively. The symmetrical triangular-shaped membership function is selected for all of variables.

Defuzzification is a mapping from the space of fuzzy control action into the space of crisp control action defined over the output universe of discourse. The *centroid* defuzzification strategy is employed.

2) Rule Base and Decision-Making

A rule base consists of a set of fuzzy conditional statements. This paper builds up a complete but simple rule base which includes 49 rules. Table 1 displays all of them, where the first row lists all of seven linguistic values of the error and the first

column is the set of linguistic values of the change of error, while other elements are corresponding fuzzy control actions.

Table 1 Rule base of FL sub-controller

	NB	NM	NS	Z	PS	PM	PB
NB	NB	NB	NB	NB	NM	Z	Z
NM	NB	NB	NB	NB	NM	Z	Z
NS	NM	NM	NM	NM	Z	PS	PS
Z	NM	NM	NS	Z	PS	PM	PM
PS	NS	NS	Z	PM	PM	PM	PM
PM	Z	Z	PM	PB	PB	PB	PB
PB	Z	Z	PM	PB	PB	PB	PB

Fuzzy decision-making is based on the human approximate reasoning. Here, the *sup-min* operation by Zadeh is used in the compositional rule of inference.

3). GA based Optimal Design of Scaling Factors

The selection of scaling factors affects not only the transient and steady-state performances of controlled system, but also its stability in extreme cases. In the past years, the input scaling factors were usually designed by trial-and-error with the aid of expert's knowledge and designer's experience. It is clear that this is quite time consuming and moreover the set of their optimal values could not be found. Hence, the effective method to design scaling factors and their optimization should receive attentions of researchers. Although many efforts have been made to optimize membership functions and control rules using GAs, little literature was devoted to the optimization of scaling factors. In this work, a GA is used to find the optimal values of two input scaling factors.

GAs are based on the mechanics of natural selection and natural genetics. In order to make GA work, the natural parameter set (k_e, k_c) has to be coded as a string of characters with a finite length. The string of characters could be a binary number, a integer, a floating-point number, a symbol, or a matrix. Here the floating-point number representation is selected. This is because when the parameters themselves are floating-point numbers, the floating-point number representation avoids encoding and decoding process. Moreover, it can make the GA work more efficient and converge faster.

Typically, A GA operates by simply repeating the following 4 steps of a cycle:

(1). creation of population.

(2). evaluation of elements.

(3). selection of best individuals

(4). genetic manipulation between each pair of parents

To evaluate the individuals of population in each generation, a fitness function must be specified. The performance criterion "ITAE" shown in equation (1) is here used for this purpose. In genetic manipulation, two of stochastic operators "crossover" and "mutation" are utilized.

$$ITAE = \int t|e|dt \qquad (1)$$

For the position servo system in section 5, each generation has a population of 40 individuals. After 25 iterations, the optimal set of input scaling factors are obtained.

4. Optimization of PI Sub-Controller Using GA

There are many methods to design PI controllers. Here, GA is again used to tune the PI sub-controller.

To get a global optimal result, GA search for PI sub-controller design is carried out based on the overall configuration given in Fig 1. The ITAE performance criterion is still used as the fitness function, and the natural parameter set (K_p, T_i) is also coded as a floating-point number.

Again considering the position servo system in section 5, the best parameters of PI sub-controller are attained also in the 25th generation.

5. Application to Position Control of Motor Servo System

The proposed concurrent FL-PI controller is applied to the position control of a DC motor servo system to illustrate its feasibility and effectiveness. Fig 3 shows the structure of system, where θ is the angular position of output shaft, ω is the angular speed of output shaft, θ° is the command input of angular position and ω° is the command input of angular speed. A FANUC-B8 permanent magnet DC motor is used as the testbed. The dynamic model of the motor with its power driver without consideration of load disturbance can be represented by the transfer function as follows:

$$G(s) = \frac{77}{s(0.0038s + 1) + 77} \tag{2}$$

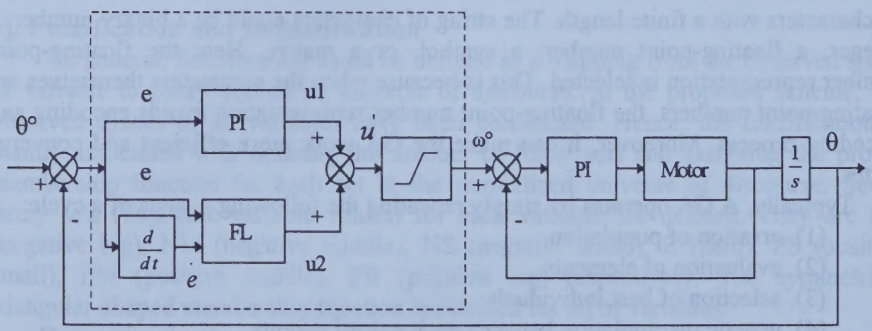


Fig 3 Structure of a DC motor position servo system

The system has two loops: a speed control loop and a position control loop. The proposed concurrent controller is used as the position controller, while the speed controller employs a conventional PI controller. The design of the concurrent position controller has been discussed above. The speed controller is also devised by using same GA aiming at minimization of performance index "ITAE". The output scaling

factor of FL sub-controller in the position loop corresponds to the maximum angular speed of output shaft.

The simulation investigation is carried out by means of SIMULINK facility. Table 1 lists the optimal set of concurrent position controller parameters designed using Gas. Fig 4 shows the angular position response of servo system to a step input. Fig 5 and Fig 6 are the angular position responses of system to the same step input when replacing the concurrent position controller with a single FL controller and a single PI controller respectively. It can be seen that the closed loop system using concurrent controller has a settling time which is only a half of that the system employing a single FL controller has. Moreover, the merit of the FL controller without overshoot is almost maintained. On the other hand, the system utilizing the concurrent control scheme hardly has any overshoot while the single PI controller controlled system has it. More important advantage is the servo system using proposed concurrent controller has better robustness than single PI controller controlled system. This is the ongoing work of authors of the paper.

Table 1 Optimal parameters of concurrent position controller

K_e	K_c	K_u	K_p	T_i
0.8905	0.0377	20	100.6	1710

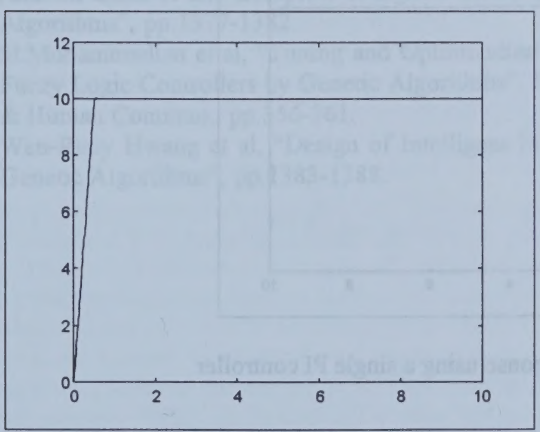


Fig 4 Position response using concurrent FL-PI controller

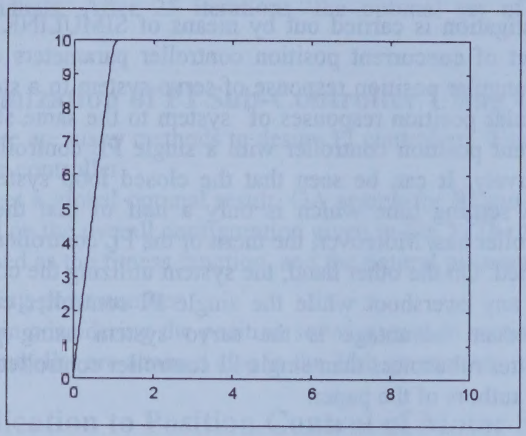


Fig 5 Position response using a single FL controller

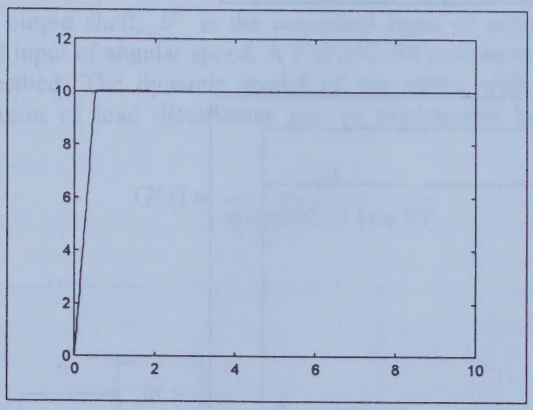


Fig 6 Position response using a single PI controller

6. Conclusion

A novel scheme of hybrid controller -- concurrent FL-PI controller -- is proposed. The input scaling factors of FL sub-controller and parameters of PI sub-controller are designed using GAs. The proposed scheme is applied to the position control of a motor servo system. Simulation results show that the concurrent FL-PI controller is superior to both the single FL controller and single PI controller. Moreover, unlike most of hybrid schemes, the concurrent controller does not switch between sub-controllers, and therefore it does not produce annoying switching chattering.

Reference

[1]. E. H. Mamdani et al, "An Experiment in Linguistic Synthesis with Fuzzy Logic Controller", Int. Jour. Man Mach. Studies, vol.7, pp.1-13,1975.

[2]. L. A. Zadeh, "Fuzzy Set", Info. & Contr., vol.8, pp.338-353, 1985.

[3]. Chuen Chien Lee, "Fuzzy Logic in Control Systems: Fuzzy Logic Controller --- Part 1, Part 2", IEEE Trans. on Sys., Man, and Cyber.,Vol. 20, No.2, pp.404-435, Mar. 1990.

[4]. B. da Silva et al, "Real Time Fuzzy Adaptive Controller for an Four Asymmetric Quadrant Power Converter", Conf. Rec., IEEE IAS Ann. Meeting, pp.872-878, 1987.

[5]. Y. F. Li et al, "Development of Fuzzy Algorithm for Servo Systems", IEEE Control System Magazine, April, 1989.

[6]. Chung-Yuen Won et al, "Robust Position Control of Induction Motor Using Fuzzy Logic Control", pp. 472-481.

[7]. Feng-Fu Cheng et al, "Application of Fuzzy Logic in the Speed Control of AC Servo System and an Intelligent Inverter", IEEE Trans. on Energy Conver., vol. 8, No. 2, pp.312-317, June 1993.

[8]. Jong-Hwan Kim et al, "Fuzzy Precompensated PID Controllers", IEEE Trans. on Contr. Sys. Tech., Vol.2, No.4, pp.406-411, Dec. 1994.

[9]. N. Matsunaga et al, "Fuzzy Hybrid Control of DC Servomotor", Elec. Engr. Japan, Vol.111, No.6, pp.105-111, 1991.

[10]. Jindrich Liska et al, "Complete Design of Fuzzy Logic Systems Using Genetic Algorithms", pp.1377-1382.

[11]. M.Mohammadian et al, "Tuning and Optimization of Membership Functions of Fuzzy Logic Controllers by Genetic Algorithms", IEEE Int. Workshop on Robot & Human Commun., pp.356-361.

[12]. Wen-Ruey Hwang et al, "Design of Intelligent Fuzzy Logic Controllers Using Genetic Algorithms", pp.1383-1388.

Application of Peak Pattern Technique to Design Fuzzy Controllers to Meet Closed-Loop Fuzzy Specifications

Cezary Kolodziej and Roland Priemer
ckolodzi@eecs.uic.edu priemer@eecs.uic.edu
Electrical Engineering and Computer Science Department
The University of Illinois at Chicago
Chicago, Illinois

Abstract

Usually, fuzzy controller design is based on expert knowledge of plant input/output behavior.Very often the control knowledge of the expert is not complete or, even worse, it consists of contradictory rules. Thus, the performance of the closed-loop system often does not meet requirements. We give a theorem about obtaining the closed-loop system fuzzy relation using the fuzzy relation of the controller and the fuzzy relation of the plant. Based on this theorem and using a fuzzy rule model of the plant and a desired fuzzy rule model of the closed-loop system, we present a new technique to synthesize the fuzzy controller. Using the peak pattern technique, we develop a complete design procedure to obtain the fuzzy controller, and then apply it to an example showing that the given technique is easy to apply.

1 Introduction

Since Zadeh's first article on fuzzy sets in 1965 [19], many researchers have contributed to the development of fuzzy set theory. The pioneering research of Mamdani and his colleagues [9, 10] coined fuzzy control, a new approach in control theory [11]. For the last 20 years, fuzzy control theory has been applied to a wide variety of control problems [1, 4, 5, 6, 16, 17, 18].

Despite its strengths, fuzzy control theory has limitations. Because the design of a fuzzy controller is based on control knowledge of an expert, the essential part of the fuzzy controller is a set of linguistic control rules in the form of *IF-THEN* statements related by the dual concepts of fuzzy implication and the compositional rule of inference [12]. This form of the fuzzy controller is not well suited for analysis and synthesis of fuzzy closed-loop systems.

Usually, fuzzy controller design is based on ad hoc techniques, because at present there are no widely accepted systematic procedures for the design of fuzzy controllers. Some design methodologies are discussed in [8, 3, 18, 16, 13]. Almost none of them use predefined performance criteria to design a fuzzy controller. Trial and error methods are used to achieve acceptable performance. Often, the problems with tuning a fuzzy controller are related to contradictory information or incomplete control knowledge in the fuzzy rule base of the fuzzy controller. Some design methodologies use prespecified

performance objectives [3, 7]. However, they are based on fuzzy adaptive algorithms that require much computation and are not known to be stable nor globally convergent. Here, we give a design procedure of a fuzzy controller based on fuzzy closed-loop system specifications inherently included in the fuzzy rules of a closed-loop system.

In Section 2 we present a theorem of minimum-order fuzzy closed-loop system relation calculation. Based on this theorem, in Section 3 we give a synthesis algorithm for a fuzzy controller. We implement a peak pattern technique to find a fuzzy controller to make it exhibit major features of performance objectives. In section 4 we give an example using the synthesis algorithm developed in Section 3.

2 Minimum-order relation of a closed-loop system

We consider a class of systems that can be represented by the block diagram shown in Figure 1. It consists of two major blocks. The block "Process" is a fuzzy model of a

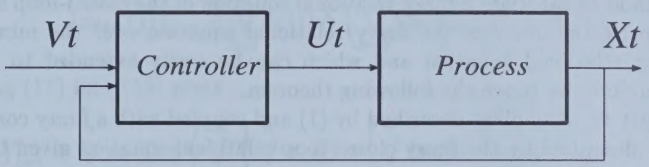


Figure 1: Block diagram of a fuzzy closed-loop system.

plant to be controlled. To calculate the output X_{t+1}, given X_t, U_t and the process fuzzy relation R_p, the compositional rule of inference is used [12] as follows

$$X_{t+1} = X_t \circ U_t \circ R_p, \; R_p \in F\left(\bar{X} \times \bar{U} \times \bar{X}\right) \tag{1}$$

where $X_t \in F\left(\bar{X}\right)$ and $X_{t+1} \in F\left(\bar{X}\right)$ are fuzzy state variables at times t and $t+1$, respectively, $U_t \in F\left(\bar{U}\right)$ is the plant input, $\bar{X}$ and $\bar{U}$ are universes of discourse, and $\circ$ is the max-min composition operator.

The block "Controller" can be described by the fuzzy relational equation

$$U_t = X_t \circ V_t \circ R_c \tag{2}$$

where $R_c = F \in \left(\bar{X} \times \bar{V} \times \bar{U}\right)$ is the fuzzy relation of the controller and $V_t \in F\left(\bar{V}\right)$ is an external input fuzzy variable.

Very little work concerned with the analysis of a fuzzy closed-loop control system has been reported. Tong [15] has studied a fuzzy closed-loop system composed of a process and a fuzzy controller described by (1) and (2). He proved that the closed-loop system can be described by

$$X_{t+1} = X_t \circ (X_t \circ V_t \circ R_c) \circ R_p \tag{3}$$

However, Equation (3) has a higher order than we would expect according to Figure 1. The increased order of the fuzzy relational equation of a fuzzy closed-loop system is a

major drawback of the fuzzy description. For these reasons Chen *at el* [2] have proposed an analytical model of a fuzzy closed-loop control system given by

$$X_{t+1} = X_t \circ R \tag{4}$$

The model they proposed does not increase the dimension of the closed-loop fuzzy relation, and it is based on the linguistic fuzzy model of the process to be controlled and on the fuzzy relation of a fuzzy controller. Chen *at el* [2] also obtained the closed-loop fuzzy relation using

$$R = \bigvee_{i=1}^{n} \{[A_i \wedge (R_c \circ B_i)] \wedge C_i\} \ , \ R \in F\left(\bar{X} \times \bar{X}\right) \tag{5}$$

Equation (5) was derived just for the particular case when there is no external input to the fuzzy controller, and the linguistic model of the plant has the particular form consisting of two fuzzy propositions in the antecedent(A_i, B_i) and a single fuzzy proposition in the consequent of a fuzzy rule (C_i). This limits the applicability of the method not only to error based control strategies but also to very simple (low-order) plants. We would prefer to have a method to calculate a fuzzy relational equation of the closed-loop system which does not increase the order of the fuzzy relational equation over the minimum-order-required fuzzy relational equation and which can be easily extended to higher order systems. Therefore, we prove the following theorem.

Theorem : Any plant described by (1) and coupled with a fuzzy controller given by (2) can be described by the fuzzy closed-loop relational equation given by

$$X_{t+1} = X_t \circ V_t \circ R_l \tag{6}$$

where the matrix $R_l \in F\left(\bar{X} \times \bar{V} \times \bar{X}\right)$ is a closed-loop fuzzy relation calculated as follows

$$R_l(i,j,l) = \bigvee_{k=1}^{m} \{R_c(i,k,l) \wedge R_p(i,j,k)\} \tag{7}$$

The $R_l(i,j,l)$ are elements of the 3-dimensional closed-loop system fuzzy relation, and m is the number of discrete points defined in $\bar{U}$.

Proof : Suppose a fuzzy system is described by the fuzzy relation (1) where X_t and X_{t+1} are given by $X_t = \{x_t(i)\}_{i=1}^{n}$ and $X_{t+1} = \{x_{t+1}(j)\}_{j=1}^{n}$, respectively, and n is the number of discrete points defined in $\bar{X}$. U_t is given by $U_t = \{u_t(k)\}_{k=1}^{m}$, where m is the number of discrete points defined in $\bar{U}$. R_p is a 3-dimensional fuzzy relation of a fuzzy process.

The algorithm to calculate X_{t+1} is given by

$$X_{t+1} = X_t \circ \left(\bigvee_{k=1}^{m} u_t(k) \wedge R_p(i,j,k)\right) \tag{8}$$

By further extension, we get

$$X_{t+1} = \bigvee_{i=1}^{n} x_t(i) \wedge \left(\bigvee_{k=1}^{m} u_t(k) \wedge R_p(i,j,k)\right) \tag{9}$$

A 3-dimensional fuzzy controller description has the form given by (2). To have the same scale as for X_t and X_{t+1} (Since we describe the output of the closed-loop control system in terms of X_t and X_{t+1}, we want to use the same scale to describe V_t, and then be able to compare it with X_t.) we describe V_t similarly to X_t and X_{t+1}. Thus, $V_t = \{v_t(l)\}_{l=1}^{n}$.

Using (2), we can calculate U_t with

$$\{u_{t+1}(k)\}_{k=1}^{m} = X_t \circ \left\{ \bigvee_{l=1}^{n} \left(v_t(l) \wedge R_c(i,k,l) \right) \right\} \tag{10}$$

By further extension, we can write

$$\{u_{t+1}(k)\}_{k=1}^{m} = \bigvee_{i=1}^{n} \left[x_t(i) \wedge \left\{ \bigvee_{l=1}^{n} \left(v_t(l) \wedge R_c(i,k,l) \right) \right\} \right] \tag{11}$$

When we connect all closed-loop system components together, we obtain the closed-loop fuzzy relation given by (3). Using the extended notation, (3) can be rewritten to become

$$\{x_{t+1}(j)\}_{j=1}^{n} = X_t \circ \left(\bigvee_{k=1}^{m} \{u_t(k) \wedge R_p(i,j,k)\} \right) \tag{12}$$

Thus, we get

$$\{x_{t+1}(j)\}_{j=1}^{n} = \bigvee_{i=1}^{n} x_t(i) \wedge \left(\bigvee_{k=1}^{m} \{u_t(k) \wedge R_p(i,j,k)\} \right) \tag{13}$$

Substituting (11) into (13) gives

$$\{x_{t+1}(j)\}_{j=1}^{n} = \bigvee_{i=1}^{n} x_t(i) \wedge \left(\bigvee_{k=1}^{m} \left\{ \bigvee_{i=1}^{n} x_t(i) \wedge \left[\bigvee_{l=1}^{n} \left(v_t(l) \wedge R_c(i,k,l) \right) \right] \wedge R_p(i,j,k) \right\} \right) \tag{14}$$

By switching the second and third max-operators, we get

$$\{x_{t+1}(j)\}_{j=1}^{n} = \bigvee_{i=1}^{n} x_t(i) \wedge \left(\bigvee_{i=1}^{n} x_t(i) \wedge \bigvee_{k=1}^{m} \left[\bigvee_{l=1}^{n} \left(v_t(l) \wedge R_c(i,k,l) \right) \right] \wedge R_p(i,j,k) \right) \tag{15}$$

This simplifies to

$$\{x_{t+1}(j)\}_{j=1}^{n} = \bigvee_{i=1}^{n} x_t(i) \wedge \left(\bigvee_{k=1}^{m} \left[\bigvee_{l=1}^{n} \left(v_t(l) \wedge R_c(i,k,l) \right) \right] \wedge R_p(i,j,k) \right) \tag{16}$$

Further, we can write

$$\{x_{t+1}(j)\}_{j=1}^{n} = \bigvee_{i=1}^{n} x_t(i) \wedge \left(\bigvee_{l=1}^{n} v_t(l) \wedge \left[\bigvee_{k=1}^{m} \{\wedge R_c(i,k,l) \wedge R_p(i,j,k)\} \right] \right) \tag{17}$$

On the other hand, we know that the fuzzy closed-loop system can be defined by the fuzzy relation given by

$$X_{t+1} = X_t \circ V_t \circ R_l \tag{18}$$

where $R_l = F \in \left(\bar{X} \times \bar{V} \times \bar{X} \right)$ is the closed-loop system fuzzy relation. Using the extended notation, (18) becomes

$$\{x_{t+1}(j)\}_{j=1}^{n} = X_t \circ \left(\bigvee_{l=1}^{n} \{v_t(l) \wedge R_l(i,j,l)\} \right) \tag{19}$$

Thus, we get

$$\{x_{t+1}(j)\}_{j=1}^{n} = \bigvee_{i=1}^{n} x_t(i) \wedge \left(\bigvee_{l=1}^{n} \{v_t(l) \wedge R_l(i,j,l)\} \right) \tag{20}$$

If we compare (20) with (17), we get the final result

$$R_l(i,j,l) = \bigvee_{k=1}^{m} \{R_c(i,k,l) \wedge R_p(i,j,k)\} \tag{21}$$

Graphically, we can visualize the closed-loop system fuzzy relation calculation as that of obtaining the minimum overlapping of two 3-dimensional arrays through U projections in each component of the fuzzy relation. Hence, it can be easily extended to higher order systems.

3 Synthesis of a controller using the peak pattern technique

Given a fuzzy model of a plant to be controlled in terms of rules such as

$$\begin{array}{lllllll} p_1: & If & X_t \text{ is } A_1 & and & U_t \text{ is } B_1 & then & X_{t+1} \text{ is } C_1 \\ p_2: & If & X_t \text{ is } A_2 & and & U_t \text{ is } B_2 & then & X_{t+1} \text{ is } C_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ p_n: & If & X_t \text{ is } A_n & and & U_t \text{ is } B_n & then & X_{t+1} \text{ is } C_n \end{array} \tag{22}$$

where p_i is the ith fuzzy rule, $X_t \in F\left(\bar{X}\right)$, $X_{t+1} \in F\left(\bar{X}\right)$, and $U_t \in F\left(\bar{U}\right)$ are defined as before, $\bar{X}$ and $\bar{U}$ are universes of discourse, A_i and B_i are linguistic values in fuzzy propositions of the antecedents of the rules, and the C_i are linguistic values in fuzzy propositions of the consequents of the rules. Fuzzy rules describing dynamic behavior of the closed-loop system are given by

$$\begin{array}{lllllll} cl_1: & If & X_t \text{ is } A_1 & and & V_t \text{ is } D_1 & then & X_{t+1} \text{ is } C_1 \\ cl_2: & If & X_t \text{ is } A_2 & and & V_t \text{ is } D_2 & then & X_{t+1} \text{ is } C_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ cl_n: & If & X_t \text{ is } A_n & and & V_t \text{ is } D_n & then & X_{t+1} \text{ is } C_n \end{array} \tag{23}$$

where $V_t \in F\left(\bar{V}\right)$ is defined as before and the D_i are linguistic values. We want to synthesize a fuzzy controller described in terms of rules such as

$$\begin{array}{lllllll} c_1: & If & X_t \text{ is } A_1 & and & V_t \text{ is } D_1 & then & U_t \text{ is } B_1 \\ c_2: & If & X_t \text{ is } A_2 & and & V_t \text{ is } D_2 & then & U_t \text{ is } B_2 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ c_m: & If & X_t \text{ is } A_m & and & V_t \text{ is } D_m & then & U_t \text{ is } B_m \end{array} \tag{24}$$

where X_t, V_t and U_t are defined as before. Since the synthesis algorithm is based on the fuzzy rules of the closed-loop system, we have to find a fuzzy relation of a fuzzy controller which together with the fuzzy relation of a plant can compose the fuzzy relation of the closed-loop system. Unfortunately, it is a well-known fact that every composition operation of two fuzzy sets/relations gives a more fuzzy set/relation than its "ancestors" [14]. Thus, the chance that we would get an exact solution of a fuzzy relational equation is low. Therefore, we implement a peak pattern technique to find a fuzzy controller that makes the closed-loop system exhibit major features of performance objectives specified in the closed-loop system fuzzy rules.

Our method of design assumes that the major feature of a fuzzy set is mostly described by its maximum membership function. Based on that, we state [14] that : two fuzzy sets X and Y are 'equivalent' if they have the same peak pattern.

A peak pattern, P, of a fuzzy set A on a finite universe of discourse X is a binary map of the discrete membership function of A, μ_A, with

$$\begin{aligned} P_A(x_i) &= 0 \; if \; \mu_A(x_i) \neq \mu_{max} \\ &= 1 \; if \; \mu_A(x_i) = \mu_{max} \end{aligned} \tag{25}$$

where μ_{max} is the maximum value of the membership function. The peak pattern technique helps to eliminate the "fuzzifing effect" of the max-min composition, and it makes the design procedure straightforward.

The proposed design algorithm, that is based on equations (7) and (25), consists of the following steps

1. Based on the fuzzy rules of a plant, obtain the fuzzy relation of a plant R_p as

$$R_p = \bigvee_{i=1}^{p_n} \{X_{t_i} \wedge U_{t_i} \wedge X_{t+1_i}\} \tag{26}$$

   where p_n is the number of fuzzy rules describing a plant.

2. Acquire information about performance objectives and obtain the fuzzy rules of a closed-loop system.

3. Based on the fuzzy rules of the closed-loop system, obtain the fuzzy relation of a closed-loop system R_l as

$$R_l = \bigvee_{i=1}^{cl_n} \{X_{t_i} \wedge V_{t_i} \wedge X_{t+1_i}\} \tag{27}$$

   where cl_n is the number of fuzzy rules describing a closed-loop system.

4. Obtain the peak pattern of the fuzzy relation R_p of the plant and obtain the peak pattern of the fuzzy relation R_l of the closed-loop system.

5. Initialize the fuzzy relation R_c of the controller with grades of membership values representing "nothing".

6. Do the following for all elements in the fuzzy relation of the initialized fuzzy controller, i.e.

$$\begin{aligned} \text{if } \{R_l(i,j,l) \wedge R_p(i,j,k)\} &= 1 \quad \text{then} \quad R_c(i,j,k) = R_l(i,j,l) \\ &\quad \text{otherwise} \quad R_c(i,j,k) \text{ is not changed} \end{aligned} \tag{28}$$

7. Based on the synthesized peak pattern of the fuzzy controller relation, obtain the fuzzy rules of the fuzzy controller.

The design procedure inherently includes closed-loop system specifications captured in the fuzzy rules of a closed-loop system.

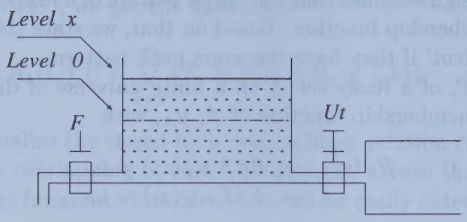


Figure 2: Water container.

4 Example

Consider a water container with a controllable water level x as shown in Figure 2. The water container is described by fuzzy rules listed below.

p_1 : if X_t is NB and U_t is PB then X_{t+1} is Z
p_2 : if X_t is NB and U_t is PS then X_{t+1} is NS
p_3 : if X_t is NB and U_t is Z then X_{t+1} is NB
p_4 : if X_t is NB and U_t is NS then X_{t+1} is NB
p_5 : if X_t is NB and U_t is NB then X_{t+1} is NB
p_6 : if X_t is NS and U_t is PB then X_{t+1} is PS
p_7 : if X_t is NS and U_t is PS then X_{t+1} is Z
p_8 : if X_t is NS and U_t is Z then X_{t+1} is NS
p_9 : if X_t is NS and U_t is NS then X_{t+1} is NB
p_{10} : if X_t is NS and U_t is NB then X_{t+1} is NB
p_{11} : if X_t is Z and U_t is PB then X_{t+1} is PB
p_{12} : if X_t is Z and U_t is PS then X_{t+1} is PS
p_{13} : if X_t is Z and U_t is Z then X_{t+1} is Z
p_{14} : if X_t is Z and U_t is NS then X_{t+1} is NS
p_{15} : if X_t is Z and U_t is NB then X_{t+1} is NB
p_{16} : if X_t is PS and U_t is PB then X_{t+1} is PB
p_{17} : if X_t is PS and U_t is PS then X_{t+1} is PB
p_{18} : if X_t is PS and U_t is Z then X_{t+1} is PS
p_{19} : if X_t is PS and U_t is NS then X_{t+1} is Z
p_{20} : if X_t is PS and U_t is NB then X_{t+1} is NS
p_{21} : if X_t is PB and U_t is PB then X_{t+1} is PB
p_{22} : if X_t is PB and U_t is PS then X_{t+1} is PB
p_{23} : if X_t is PB and U_t is Z then X_{t+1} is PB
p_{24} : if X_t is PB and U_t is NS then X_{t+1} is PS
p_{25} : if X_t is PB and U_t is NB then X_{t+1} is Z

where PB means positive big, PS means positive small, Z means zero, NS means negative small, and NB means negative big. The fuzzy sets representing these fuzzy values are given Table 1. Let us assume that an expert has to design a fuzzy controller to make a closed-loop system follow the fuzzy rules listed below.

error	-2	-1	0	1	2
PB	0	0	0	0.5	1
PS	0	0	0.5	1	0
Z	0	0.5	1	0.5	0
NS	0	1	0.5	0	0
NB	1	0.5	0	0	0

control variable	-4	-3	-2	-1	0	1	2	3	4
PB	0	0	0	0	0	0	0	0.5	1
PS	0	0	0	0	0	0.5	1	0.5	0
Z	0	0	0	0.5	1	0.5	0	0	0
NS	0	0.5	1	0.5	0	0	0	0	0
NB	1	0.5	0	0	0	0	0	0	0

Table 1: Fuzzy labels of the system linguistic variable.

cl_1 : if X_t is NB and V_t is PB then X_{t+1} is Z
cl_2 : if X_t is NB and V_t is PS then X_{t+1} is Z
cl_3 : if X_t is NB and V_t is Z then X_{t+1} is Z
cl_4 : if X_t is NB and V_t is NS then X_{t+1} is NS
cl_5 : if X_t is NB and V_t is NB then X_{t+1} is NB
cl_6 : if X_t is NS and V_t is PB then X_{t+1} is PS
cl_7 : if X_t is NS and V_t is PS then X_{t+1} is PS
cl_8 : if X_t is NS and V_t is Z then X_{t+1} is Z
cl_9 : if X_t is NS and V_t is NS then X_{t+1} is NS
cl_{10} : if X_t is NS and V_t is NB then X_{t+1} is NB
cl_{11} : if X_t is Z and V_t is PB then X_{t+1} is PB
cl_{12} : if X_t is Z and V_t is PS then X_{t+1} is PS
cl_{13} : if X_t is Z and V_t is Z then X_{t+1} is Z
cl_{14} : if X_t is Z and V_t is NS then X_{t+1} is NS
cl_{15} : if X_t is Z and V_t is NB then X_{t+1} is NB
cl_{16} : if X_t is PS and V_t is PB then X_{t+1} is PB
cl_{17} : if X_t is PS and V_t is PS then X_{t+1} is PS
cl_{18} : if X_t is PS and V_t is Z then X_{t+1} is Z
cl_{19} : if X_t is PS and V_t is NS then X_{t+1} is NS
cl_{20} : if X_t is PS and V_t is NB then X_{t+1} is NS
cl_{21} : if X_t is PB and V_t is PB then X_{t+1} is PB
cl_{22} : if X_t is PB and V_t is PS then X_{t+1} is PS
cl_{23} : if X_t is PB and V_t is Z then X_{t+1} is Z
cl_{24} : if X_t is PB and V_t is NS then X_{t+1} is Z
cl_{25} : if X_t is PB and V_t is NB then X_{t+1} is Z

Using the synthesis algorithm presented in Section 3 we obtain the set of fuzzy rules of a fuzzy controller. The rules are listed below.

c_1 : if X_t is NB and V_t is PB then U_t is PB
c_2 : if X_t is NB and V_t is PS then U_t is PB
c_3 : if X_t is NB and V_t is Z then U_t is PB
c_4 : if X_t is NB and V_t is NS then U_t is PS
c_5 : if X_t is NB and V_t is NB then U_t is NB or NS or Z
c_6 : if X_t is NS and V_t is PB then U_t is PB
c_7 : if X_t is NS and V_t is PS then U_t is PB
c_8 : if X_t is NS and V_t is Z then U_t is PS
c_9 : if X_t is NS and V_t is NS then U_t is Z
c_{10} : if X_t is NS and V_t is NB then U_t is NB or NS
c_{11} : if X_t is Z and V_t is PB then U_t is PB
c_{12} : if X_t is Z and V_t is PS then U_t is PS
c_{13} : if X_t is Z and V_t is Z then U_t is Z
c_{14} : if X_t is Z and V_t is NS then U_t is NS
c_{15} : if X_t is Z and V_t is NB then U_t is NB
c_{16} : if X_t is PS and V_t is PB then U_t is PB or PS
c_{17} : if X_t is PS and V_t is PS then U_t is Z
c_{18} : if X_t is PS and V_t is Z then U_t is NS
c_{19} : if X_t is PS and V_t is NS then U_t is NB
c_{20} : if X_t is PS and V_t is NB then U_t is NB
c_{21} : if X_t is PB and V_t is PB then U_t is PB or PS or Z
c_{22} : if X_t is PB and V_t is PS then U_t is NS
c_{23} : if X_t is PB and V_t is Z then U_t is NB
c_{24} : if X_t is PB and V_t is NS then U_t is NB
c_{25} : if X_t is PB and V_t is NB then U_t is NB

As expected the synthesized fuzzy controller is not unique. There is a set of fuzzy controllers that meet closed-loop system specifications. Which one to choose is a question that has not been completely answered yet.

5 Conclusions

A new synthesis method for a fuzzy controller has been presented. The method uses a peak pattern technique to eliminate the "fuzzifying effect" of the max-min composition and makes the closed-loop system exhibit major features of performance objectives specified in the closed-loop system fuzzy rules. It speeds up the design procedure by eliminating the time that is required to obtain a control algorithm from an expert. It inherently avoids unnecessary, sometimes even contradictory, expert information.

References

[1] A. Bardossy and L. Duckstein. *Fuzzy Rule-Based Modelling with Application to Geophysical, Biological and Engineering Systems.* CRC Press, New York, 1995.

[2] J. Chen and L. Chen. Study on stability of closed-loop control systems. *Fuzzy Sets and Systems*, 57:159–168, 1993.

[3] S.Z. He and P. Z. Wang. Control of dynamical processes using an on-line rule-adaptive fuzzy control system. *Fuzzy Sets and Systems*, 54:11–22, 1993.

[4] H. Hellendoorn and R. Palm. Fuzzy systems technologies at siemens r & d. *Fuzzy Sets and Systems*, 63:245–269, 1994.

[5] M.A. Iskandar, Y. Mizutani, A. Suzuki, and M. Ishizeki. Stabilizing control of power system usig fuzzy control. *Electrical Engineering in Japan*, 114(3):33–45, 1994.

[6] P.J. King and E.H Mamdani. The application of fuzzy control systems to industrial processes. *Automatica*, 3:235–242, 1977.

[7] J.R. Layne and K.M. Passino. Fuzzy model reference learning control for cargo ship steering. *IEEE Transactions on Control Systems*, December:23–34, 1993.

[8] C.C. Lee. Fuzzy logic in control systems : Fuzzy logic controller part 1 and part2. *IEEE Transactions on Systems, Man, and Cybernetics*, 20(2):404–418, 419–436, 1990.

[9] E.H. Mamdani. Applications of fuzzy algorithm for simple dynamic plant. *Proceedings IEE*, 121(12):1585–1588, 1974.

[10] E.H. Mamdani. Applications of fuzzy logic to approximate reasoning using linguistic synthesis. *IEEE Transactions on Systems, Man, and Cybernetics*, 26(12):1182–1191, 1977.

[11] E.H. Mamdani. Twenty years of fuzzy control: axperiences gained and lessons learnt. *IEEE*, pages 339–344, 1993.

[12] W. Pedrycz. *Fuzzy Control and Fuzzy Systems.* John Wiley & Sons Inc., New York, 1993.

[13] R. Stenz and U. Kuhn. Automation of a batch distilation column using fuzzy and conventional control. *IEEE Transactions on Systems Technology*, 3(2):171–176, 1995.

[14] R.M. Tong. Analysis and control of fuzzy system using finite discrete relations. *International Journal of Control*, 27:431–440, 1978.

[15] R.M. Tong. Some properties of fuzzy feedback systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 10(6):327–330, 1980.

[16] C. von Altrock. *Fuzzy Logic and NeuroFuzzy Applications Explained.* Prentice Hall PTR, Englewood Cliffs, New Jersey 07632, 1995.

[17] T. Yamakawa. Stabilization of an inverted pendulum by a high-speed fuzzy logic controller hardware system. *Fuzzy Sets and Systems*, 32:161–180, 1989.

[18] J. Yen, R. Langari, and L.A. Zadeh. *Idustrial Applications of Fuzzy Logic and Intelligent Systems.* IEEE Press, IEEE, Inc, New York, 1995.

[19] L.A. Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

A Solution to Generalized Constant Value Control Problems of Plants with Dominant Time Delay

Guomin Li K. M. Tsang S. L. Ho
Electrical Engineering Department, Hong Kong Polytechnic University
Hung Hom, Kowloon, Hong Kong

Abstract ------ The Variable Step Approaching Digital Control Algorithm (VSADCA) is applied to the generalized constant value control problems of plants with dominant time delay. A self-tuning scheme of the algorithm is proposed. Detailed simulation investigations are carried out. A Process Trainer PT326 heating system is used to demonstrate its practical application. Both simulation and experimental results show that the self-tuning VSADCA controller is superior to the Z-N method tuned PID controller in the generalized constant value control area of plants with dominant time delay.

Keywords----- Control method Time delay Process control

1. Introduction

In recent years, many advanced control methods such as Model Reference Adaptive Control, Sliding Mode Variable Structure Control and Fuzzy Logic Control have been developed. Although they produce superior performances, these methods are usually very complicated to implement and not very suitable for real time control because they are rather time consuming. Especially, some of them have parameters which are difficult to determine. In Fuzzy Logic Control (FLC) method[1], for example, the universes of discourse of the error and the change of error, as well as the membership functions of their fuzzy sets have to be determined in advance. In practice, however, this is quite difficult and sometimes has to be based on the experiences attained previously.

On the other hand, the conventional PID controllers have still been maintaining wide applications because of their simplicity, ease of implementation and abroad effectiveness. In the tuning of PID controllers, the Ziegler and Nichols's ultimate cycle method[2] is well known and widely used in practice. However, It is found that the PID controller tuned by Z-N ultimate cycle method usually gives bigger maximum overshoot, and produces poor performances when it is applied to plants with dominant time delay.

Among the industrial practices, there is a great variety of constant value control problems, such as the constant temperature control in heating furnaces, the constant pressure control in water supplies, and the liquid level control in the tanks. In all these cases, the output variable of system is often desired to be kept at some constant level (usually called the set value or desired value) during some period and to have the maximum overshoot as small as possible in the transient progress to reach this value.

G. Li et al proposed a very simple digital control algorithm called Variable Step Approaching Digital Control Algorithm (VSADCA) for the control of plants with time delay[3]. They demonstrated that VSADCA is an outstanding control algorithm in

suppressing overshoots and is superior to PID controller tuned by Z-N ultimate cycle method in applying to time-delay dominant plants.

In this paper, based on G. Li's previous work, a self-tuning scheme of VSADCA controller for the generalized constant value control of plants with dominant time delay is developed. Detailed simulation investigations are carried out. A verifying experiment using Process Trainer PT326 heating system as the testbed is conducted. Both simulation and experimental results which are well consistent with each other not only validate the self-tuning method presented here, but also consolidate the VSADCA itself that was not verified by experiment when it was proposed.

2. VSADCA and Its Original Tuning Rule

VSADCA is a simple digital control algorithm which is easy to implement for the control of plants with time delay. It is outstanding in suppressing the overshoot and in reducing the decaying oscillations of closed loop system response. Especially, it is superior to the Z-N method tuned PID controller in the control of time-delay dominant plants. Fig 1 shows the schematic diagram of a closed loop system using VSADCA digital controller, where $y(t)$ stands for the system output, $r(t)$ denotes its

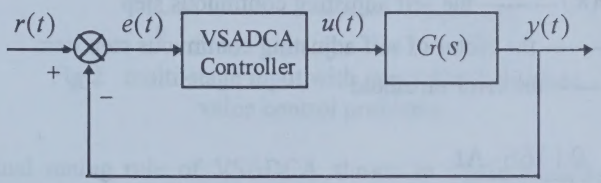


Fig 1. Schematic diagram of a closed loop system using VSADCA

set value or the system input, $e(t)$ is the error between system input and output, $u(t)$ represents the output of VSADCA controller, and $G(s)$ is the transfer function of the controlled plant with the form:

$$G(s)=\frac{k\cdot e^{-\tau s}}{Ts+1} \tag{1}$$

where

$G(s)$--------- the plant transfer function
k ---------- the static gain of plant
T ---------- the time constant of plant
τ ---------- the time delay of plant

This model can approximate a lot of real plants [4].

The mathematical expression of VSADCA and its original tuning rule are given as follows [3] :

$$u(k)=\begin{cases} u(k-1)+s_l & e(k)>T_h \text{ and } \dot{e}(k)\geq 0 \\ u(k-1)+s_s & e(k)>T_h \text{ and } \dot{e}(k)<0 \\ u(k-1)+k_i\cdot e(k) & |e(k)|\leq T_h \\ u(k-1)-s_s & e(k)<-T_h \text{ and } \dot{e}(k)>0 \\ u(k-1)-s_l & e(k)<-T_h \text{ and } \dot{e}(k)\leq 0 \end{cases} \tag{2}$$

where

$u(k)$------- the controller output at k-th sampling instant

$u(k-1)$-----the controller output at $(k-1)$-th sampling instant

$e(k)$---------the error between system input and output at k-th sampling instant

$\dot{e}(k)$-------- the derivative of error at k-th sampling instant

s_l -------- the fixed large step of controller output

s_s -------- the fixed small step of controller output

$k_i\cdot e(k)$--------- the self adjusting continuous step

k_i -------- the factor of self adjusting continuous step item

T_h ------- the error threshold

and

$$\begin{cases} s_l=\dfrac{0.1166\cdot\Delta t}{k\cdot T'\cdot\tau'} \\ s_s=\dfrac{1}{3}s_l \\ k_i=\dfrac{0.389\cdot\Delta t}{k\cdot T'\cdot\tau''} \\ u(0)=\dfrac{0.7}{k} \\ T_h=0.15 \quad (15\%\ of\ \ step\ input\ \ in\ general) \end{cases} \tag{3}$$

where

$$\begin{cases} T'=\max\{0.67,T\} \\ \tau'=\max\{0.058,\tau\} \\ \tau''=\max\{0.58,\tau\} \end{cases} \tag{4}$$

Δt -------- the sample period

$u(0)$ ----- the suggested initial value of controller output

3. Self-Tuning Scheme for Generalized Constant Value Control Problems

To keep the output of a system at an unique set-value during the whole working period is, with no doubt, a constant value control problem. In this paper, however, the concept of constant value control problem is generalized. It refers to the tracking control of a general multi-stage input with enough lasting time in each stage as formatted in Fig 2. The typical operation in the temperature control system consisting of the following commands: starting, increasing temperature, lowering temperature and stop, is an example of this kind of control problems.

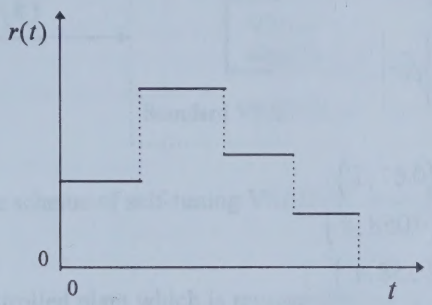


Fig 2 multi-stage input with generalized constant value control problems

The original tuning rule of VSADCA shown in equations (3) and (4) was developed from the simulation research on unit step responses of closed loop systems of a class of typical plants using VSADCA controllers [3] . Although VSADCA of equations (2) to (4) could greatly attenuate the overshoot of a closed loop controlled system, the settling time, however, may become too long when the input format is changed. Hence, it is necessary to make an adequate modification to the original tuning rule so that VSADCA can still keep the optimal control performances not only with regard to the overshoot but also in respect of the settling time when a general multi-stage input as shown in Fig 2 is applied.

The self tuning rule which is suitable for the multi-stage input, and therefore can be used in the generalized constant value control problems is proposed in the following equations:

$$\begin{cases} s_l = \dfrac{0.1166 \cdot \Delta t}{k \cdot T' \cdot \tau'} \left| \bar{\Delta r} \right| \\ s_s = \dfrac{1}{3} s_l \\ k_i = \dfrac{0.389 \cdot \Delta t}{k \cdot T' \cdot \tau''} \\ u(0) = \dfrac{0.7}{k} \bar{\Delta r} \\ T_h = 0.15 \left| \bar{\Delta r} \right| \end{cases} \tag{5}$$

where

$$\begin{cases} T' = \max\{0.67, T\} \\ \tau' = \max\{0.058, \tau\} \\ \tau'' = \max\{0.58, \tau\} \end{cases} \tag{6}$$

and

$\bar{\Delta r}$ -------- the step change of system input

It can be seen that the change of system input is used in setting the controller parameters. In other words, the controller will modify its own parameters on-line so that it fits the input changes. Although in general the self-tuning scheme to fit the input is complicated and sometimes not feasible, with the generalized constant value control problems, however, it is quite simple and very easy to carry out. The reason is that in such a system the input is a constant after each operation as shown in Fig 2, and furthermore, VSADCA is numerically implemented in the computer which can undertake many tasks at the same time.

Fig 3 shows an implementation scheme of such a self-tuning VSADCA controller. The task of the step change observer is to monitor the step change of system input and the time it happens. The parameter modification mechanism calculates and updates the VSADCA controller parameters based on equations (5) and (6) when the step change of input occurs. The purpose of the steady-state observer is to examine if the system has reached the steady state and to provide this information to the parameter modification mechanism so that it can determine whether or not the past step changes of input are also needed to be considered in the present modification. The origin injection control governs the addition of controller output "initial" value to its present value, which also happens only at the time when the step change of system input emerges.

The parameter modification is only needed when the input is step-changed. Therefore, the computation load for such a self-tuning implementation is very tiny and can be neglected.

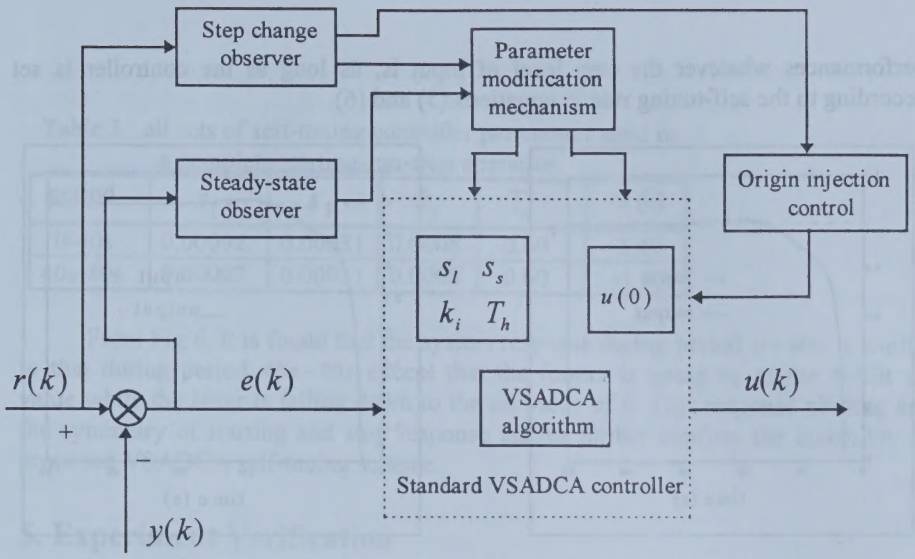


Fig 3 The scheme of self-tuning VSADCA controller

4. Case Study

Consider the controlled plant which is represented by the transfer function

$$G_1(s) = \frac{2e^{-3s}}{0.85s+1} \tag{7}$$

with a ratio of time delay to time constant $\frac{\tau}{T} \cong 3.5$. Its standard VSADCA controller parameters tuned by the original rule shown in equations (3) and (4) are tabulated in Table 1, where the sample time is set to 0.01s. The self-tuning parameters derived from the modified rule in equations (5) and (6) suitable for the case where the set value of system output is a 4-units step input are given in Table 2. The sample time

Table 1 original VSADCA controller parameters

s_l	s_s	k_i	T_h	$u(0)$
0.00023	0.00008	0.0008	0.15	0.35

Table 2 case-specific self-tuning controller parameters

s_l	s_s	k_i	T_h	$u(0)$
0.00092	0.00031	0.0008	0.60	1.40

is also set to 0.01s. The corresponding simulation results of closed loop system responses to the unit step input and 4-unit set value input are demonstrated in Fig 4 and Fig 5 respectively. It can be seen that two response curves have the same shapes, the same small overshoots, the same settling times and the same oscillation degrees. Thus, it can be concluded that the closed loop system may have the identical

performances whatever the step level of input is, as long as the controller is set according to the self-tuning rule in equations (5) and (6).

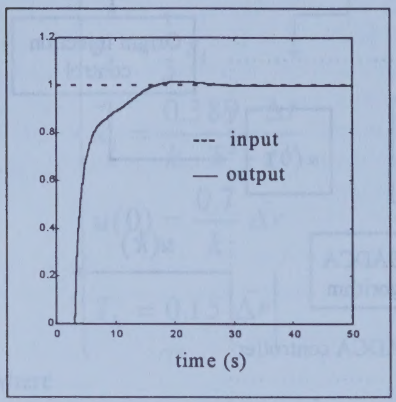


Fig 4 system response to unit step input

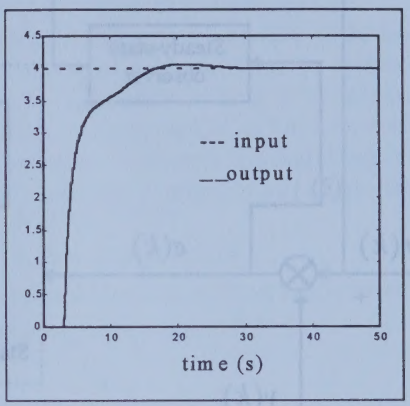


Fig 5 system response to 4-unit set value input

The more significant application of VSADCA self-tuning scheme lies in more sophisticated cases. Fig 6 gives out the plotting of system response to such a input which steps from 0 to 4 at the beginning and changes back to 0 again afterwards. This case emulates a typical complete starting-run-stop operation in the constant value control problems. With such an operation, the controller only needs to self-modify

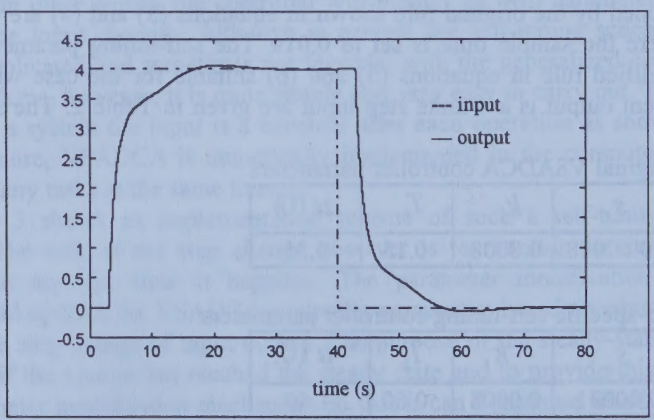


Fig 6 System response to the input with two step changes

its parameters two times, which respectively happens at the starting and stop command instants: $t = 0$ and $t = 40s$. Table 3 exhibits all sets of self-tuning controller parameters for this case, which are automatically computed at $t = 0$ and $t = 40s$ and used during two different periods: $0 - 40s$ and $40s - 80s$ respectively.

Table 3 all sets of self-tuning controller parameters used in a complete starting-run-stop operation

period	s_l	s_s	k_i	T_h	$u(0)$
0-40s	0.00092	0.00031	0.0008	0.60	1.40
40s-80s	0.00092	0.00031	0.0008	0.60	-1.40

From Fig 6, it is found that the system response during period $0-40s$ is similar to that during period $40s-80s$ except that the former is going up to the 4-unit set value while the latter is falling down to the set value of 0. This response plotting and the symmetry of starting and stop response curves further confirm the feasibility of proposed VSADCA self-tuning scheme.

5. Experiment Verification

An experiment in the laboratory is conducted to validate the VSADCA and its self-tuning scheme in practical application. The testbed is a Process Trainer PT 326 heating system from Feedback Instruments. An external time delay is added to build up a time delay dominant plant.

PT326 is a nonlinear blowing-air heating system in which the air is forced to flow along a heat-insulating pipe by a blower, the heating circuit locates near one end of the pipe and the measuring element is placed close to the other end of the pipe.

Equation (8) is an approximate transfer function expression of the heating system. Its VSADCA controller parameters calculated from self-tuning rule specific

$$G_2(s) = \frac{1.6\, e^{-1.7s}}{0.54s + 1} \tag{8}$$

to the step input with the set temperature of $45^{\circ}c$ are given in Table 4, where the sample time is set to 0.01s. To provide a comparison, the PID control experiment is also carried out. Its controller parameters tuned by Z-N ultimate cycle method are placed in Table 5, where k_p is the proportional gain of PID, T_i its integral time constant, and T_d its derivative time constant.

Table 4 parameters of VSADCA controller used in the experiment

s_l	s_s	k_i	T_h	$u(0)$
0.0025	0.0008	0.0021	0.60	1.76

Table 5 parameters of PID controller tuned by Z-N ultimate cycle method

k_p	T_i	T_d
0.481	2.142	0.5355

Fig 7 shows the simulation response and the experimental response of self-tuning VSADCA controlled closed loop system of PT326 to the step input of $45^{\circ}c$ set temperature starting at $t = 0$ instant. During the initial state, both outputs are forced to their own ambient temperature (approximately $25^{\circ}c$, which is a little different from each other because of the difficulty in emulating the actual time-variant ambient temperature). The good consistence between the simulation result and experimental result is clear. Therefore, it is confirmed that the self-tuning scheme presented in this paper and VSADCA itself are feasible in practice, and the closed loop system using VSADCA controller does have the overshoot which is suppressed greatly.

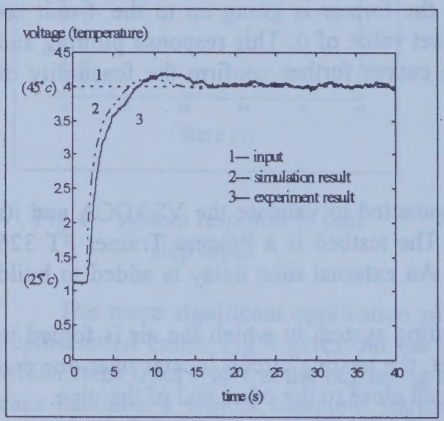


Fig 7 results of simulation and experiment for VSADCA control

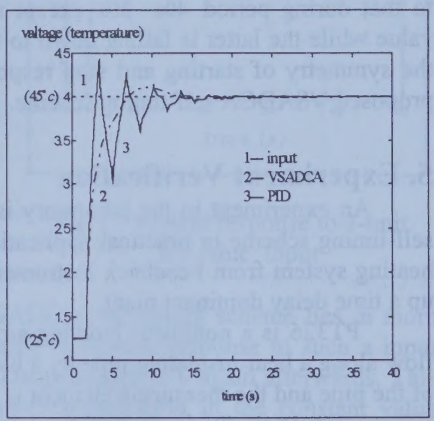


Fig 8 comparison between simulation results of VSADCA and PID

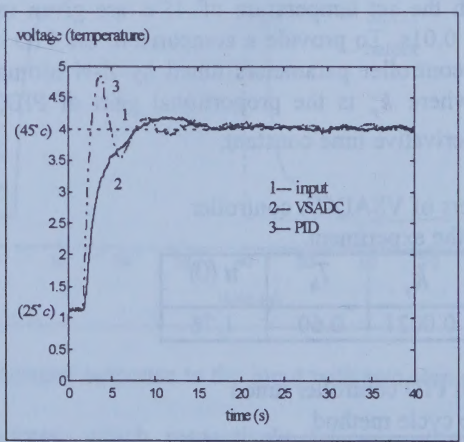


Fig 9 comparison between experiment results of VSADCA and PID

Fig 8 is the comparison of simulation results of closed loop system responses between self-tuning VSADCA controller and Z-N second method tuned PID controller. It can be seen that VSADCA controlled system has a smaller overshoot, smaller number of decaying oscillations and shorter settling time. This is verified by the succeeding comparison of experimental results shown in Fig 9. Therefore, the superiority of self-tuning VSADCA controller over Z-N second method tuned PID controller in the generalized constant value control area of plants with dominant time delay is here demonstrated.

6. Conclusion

In this paper, a simple digital control algorithm named Variable Step Approaching Digital Control Algorithm (VSADCA) is applied to the generalized constant value control problems of plants with dominant time delay. A self-tuning scheme of VSADCA is developed. Both simulation and experimental results which are well consistent with each other show that the self-tuning VSADCA controller is superior to the PID controller tuned by Z-N ultimate cycle method in the generalized constant value control area of plants with dominant time delay. Moreover, the self-tuning VSADCA controller requires only small computation effort and thus is very easy to implement in the microprocessor based real time control systems.

Reference

[1] C.C. Lee, " Fuzzy logic in control systems: Fuzzy logic controller--Part 1, Part 2", IEEE Trans. SyS. Man. Cybern., Vol. 20, No. 2, pp. 404--435, 1990.

[2] Ziegler, J. B. and Nichols, N. B., "Optimum settings for automatic controllers", Trans. American Soc. Mech. Engin. No. 64, pp. 759-768, 1942.

[3] Guomin Li, K. M. Tsang , S. L. Ho and Y. K. Wong, "Variable step approaching digital control algorithm for the control of plants with time delay", 4th Int. Conf. on Control, Automation, Robotics and Vision, Dec 3-6, 1996, Singapore.

[4] C. L. Smith, "Digital computer process control", International Company, 1972.

[5] K. M. Tsang and A. B. Rad, "A new approach to auto-tuning of PID controllers", Int. Jour. System. Sci., Vol. 26, No. 3, pp. 639-658, 1995.

An Integrated Identification and Control Procedure for Sheet and Film Processes

Andrew P. Featherstone Richard D. Braatz

Department of Chemical Engineering
University of Illinois at Urbana-Champaign
600 South Mathews Avenue, Box C-3
Urbana, IL 61801
217-333-5073 braatz@uiuc.edu

Abstract

Sheet and film processes pose a challenging identification and control problem due to high complexity, poor conditioning, and limited input-output data. The interaction between model accuracy and closed loop performance is explored for sheet and film processes using a model decomposition in terms of two static orthogonal matrices in series with a diagonal transfer function matrix. It is shown that the accuracy of the diagonal elements of the transfer function matrix directly specifies the closed loop performance achievable by a model based controller. This motivates the development of a combined identification and control procedure in which the controller is designed to be robust to model inaccuracies quantified during identification. The resulting controller is compared to an industrially accepted controller design method for a simulated blown film process.

1 Introduction

Sheet and film processes constitute a industrially important class of structured large scale systems that include coating, papermaking, and polymer film extrusion processes. Improved control of sheet and film processes can mean significant reductions in material consumption; greater production rates for existing equipment; improved product quality; elimination of product rejects; and reduced energy consumption.

The numerical difficulties associated in handling large dimensional processes prevent the application of many advanced control algorithms and identification procedures suitable for smaller scale processes [5]. More importantly, large scale systems are almost always poorly conditioned, and such processes are well-known to be difficult to control [34]. This poor conditioning is especially critical for sheet and film processes, for which the quantity of high quality experimental data is usually extremely limited [19]. These characteristics and experimental constraints lead to difficulty in obtaining acceptable models for control.

Model-based control strategies depend on process characteristics that must be accurately identified in the model, otherwise poor performance or instability may result. The purpose of this investigation is to understand the relationship between the quality of the process model and the resulting closed loop performance. Such connections between control and identification are referred to in the literature as belonging to the area of *control relevant identification.*

The manuscript is organized as follows. First, we discuss model structures appropriate for sheet and film processes, including a model decomposition in terms of two static orthogonal

matrices in series with a diagonal transfer function matrix. It is shown that the accuracy of the diagonal elements of the transfer function matrix directly specifies the closed loop performance achievable by a model based controller. This motivates the development of a combined identification and control procedure in which the controller is designed to be robust to model inaccuracies quantified during identification. The resulting controller is applied to a simulated blown film process.

2 Model Structures

The quantity of on-line data available for sheet and film processes is low, so that a model with a limited number of parameters must be selected. The traditional model structure for flat web forming processes is the Toeplitz symmetric description. For some polymer extrusion applications [27, 10], paper machines with neglected edge effects [26, 38], and adhesive coating processes [6], the circulant symmetric description can be used. This model structure has more desirable mathematical properties for the analysis of model requirements and for design of experiments [19], and is applicable to a wide class of processes of practical importance. Other researchers have chosen to parameterize process signals (e.g., process output, disturbances) and the profile response to each actuator move in terms of a linear combination of basis functions, and to identify the coefficients corresponding to each basis function. Basis functions under study include splines [21] and discrete orthonormal Chebyshev polynomials, also referred to as Gram or discrete Legendre orthogonal polynomials [24, 25].

Many process transfer function matrices can be decomposed into the *pseudo-SVD* form [19, 22]

$$P(s) = U\Lambda(s)V^T, \tag{1}$$

where U and V are constant real orthogonal matrices, and $\Lambda(s)$ is a diagonal transfer function matrix. This structure is similar to the standard singular value decomposition (SVD) [20], but the elements of the diagonal matrix $\Lambda(s)$ are transfer functions and not necessarily ordered.

A large class of industrially relevant processes can be represented in the pseudo-SVD form. For example, any plant that has the same scalar dynamics $p(s)$ between each of its manipulated variables and all the measured variables can be written as

$$P(s) = p(s)A \tag{2}$$

where A is a constant real matrix. Inserting the SVD of A and rearranging gives the pseudo-SVD form

$$P(s) = p(s)A = p(s)U\Sigma V^T = U\underbrace{(p(s)\Sigma)}_{\Lambda(s)}V^T. \tag{3}$$

There are many examples of such transfer function matrices in the chemical process industries, including simplified distillation column models [34], and every published model of sheet and film processes [26, 37, 39, 3, 36, 38, 30, 14, 23].

Processes with circulant symmetric descriptions (described above) have transfer function

matrices of the form

$$C(s) = \underbrace{\begin{bmatrix} c_1 & c_2 & \cdots & c_{m-1} & c_m & \cdots & c_m & c_{m-1} & \cdots & c_2 \\ c_2 & c_1 & c_2 & \cdots & c_{m-1} & c_m & \cdots & \ddots & \ddots & \vdots \\ \vdots & c_2 & c_1 & c_2 & \cdots & c_{m-1} & \ddots & \ddots & \ddots & c_{m-1} \\ c_{m-1} & \vdots & c_2 & c_1 & c_2 & \cdots & \ddots & \ddots & \vdots & c_m \\ c_m & c_{m-1} & \vdots & c_2 & \ddots & \ddots & \vdots & c_{m-1} & c_m & \vdots \\ \vdots & c_m & c_{m-1} & \vdots & \ddots & \ddots & c_2 & \vdots & c_{m-1} & c_m \\ c_m & \vdots & \ddots & \ddots & \cdots & c_2 & c_1 & c_2 & \vdots & c_{m-1} \\ c_{m-1} & \ddots & \ddots & \ddots & c_{m-1} & \cdots & c_2 & c_1 & c_2 & \vdots \\ \vdots & \ddots & \ddots & \cdots & c_m & c_{m-1} & \cdots & c_2 & c_1 & c_2 \\ c_2 & \cdots & c_{m-1} & c_m & \cdots & c_m & c_{m-1} & \cdots & c_2 & c_1 \end{bmatrix}}_{n \times n}, \quad (4)$$

where each c_j is a transfer function. All circulant symmetric transfer function matrices can be diagonalized by the real Fourier matrix [22, 19]

$$C(s) = R\Lambda(s)R^T. \quad (5)$$

The decomposition matrix R depends only on the process dimension and not on localized physical phenomena. This decomposition can be used for any circulant symmetric sheet and film process. An additional property that can be exploited to simplify controller synthesis is that only $(n+1)/2$ (for n odd) or $(n/2)+1$ (for n even) of the diagonal elements $\Lambda_{ii}(s)$ are distinct [22].

3 Model Requirements

One of the goals of this investigation is to gain insight as to which process characteristics must be extracted for providing adequate closed loop performance. To do this, we will assume in this section that the process can be placed in pseudo-SVD form (1), and that the orthogonal matrices U and V in (1) are exactly known. The latter assumption will be true for circulant symmetric processes (4), in which the orthogonal matrices $U = V = R$ can be computed *before experimental data is collected.* In section 4, we describe how to design controllers to be robust to inaccuracies in U and V.

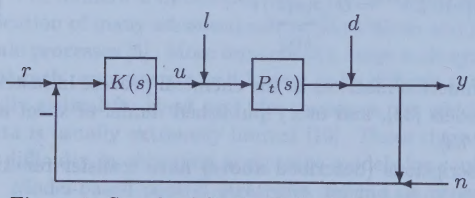


Figure 1: Standard feedback control system.

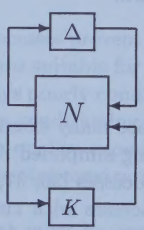


Figure 2: Uncertain system representation.

Consider a closed loop system with the standard feedback controller as shown in Figure 1. Hovd et al. have shown that for pseudo-SVD processes (1), controllers of the form

$$K(s) = V\Lambda_K(s)U^T \quad (6)$$

(referred to as *SVD controllers*) provide optimal performance for a variety of norms (e.g., H_2, H_∞), under mild technical conditions [22]. The optimality is true even for the case where model inaccuracies are represented and the performance is measured in a "worst-case" sense (see [29, 35] for thorough descriptions of how model uncertainties and worst-case performance measures are addressed). Given the relatively slow speed of measurements and manipulations compared to the speed of the rest of the process dynamics, the most important performance specification for sheet and film processes is zero steady-state error [5]. For a stable plant to have zero steady state error, the final value theorem implies that the controller must have an integrator ($1/s$) in each channel. A SVD controller with integral action in all channels while having no poles in the open right half plane will be referred to as an *integral SVD controller*. The performance and robustness characteristics of the SVD controller and the requirement of integral action motivate the following result concerning the gain identification requirements for pseudo-SVD processes.

Theorem 1 *Consider a closed loop system without actuator limitations as shown in Figure 1. Assume the true plant ($P_t(s)$) and a model of the plant ($P_m(s)$) are stable proper rational transfer functions that are of the pseudo-SVD form ($P_t(s) = U\Lambda_t(s)V^T$ and $P_m(s) = U\Lambda_m(s)V^T$). Then an integral SVD controller designed to stabilize $P_m(s)$ will also stabilize $P_t(s)$ only if $\Lambda_{t,ii}(0)/\Lambda_{m,ii}(0) > 0$, for all i. Furthermore, there exists an integral SVD controller that stabilizes both $P_m(s)$ and $P_t(s)$ and continues to do so with arbitrary detuning of its single loop gains ($\Lambda_{K,ii}(0)$) if $\Lambda_{t,ii}(0)/\Lambda_{m,ii}(0) > 0$, for all i.*

If any of the true process gains $\Lambda_{t,ii}(0)$ is equal to zero, then there does not exist a nonsingular integral controller that stabilizes the unconstrained closed loop system [12]. If none of the true process gains is exactly equal to zero, then Theorem 1 implies that an integral SVD controller exists that stabilizes both the model and the true process if each identified model gain $\Lambda_{m,ii}(0)$ has the same sign as the corresponding true process gain of the true process $\Lambda_{t,ii}(0)$. Furthermore, these sign conditions must hold for *any* model-based integral SVD controller to stabilize the true process. Relationships between Theorem 1 and other results in the literature are described elsewhere [17]. The proof of Theorem 1 is given in [17].

For a process without actuator limitations, an integral SVD controller whose design is based on a model with inaccurate signs of its steady state gains will cause an unstable closed loop response. Since sheet and film processes (like all real systems) have constraints on their actuator moves, the vector of manipulated variables for such a controller will grow in magnitude until the actuator moves become limited, irrespective of how small the disturbances are that enter the system. Although Theorem 1 does not imply that any *non-SVD* integral controller whose design is based on a model with inaccurate signs of its steady state gains will destabilize the closed loop system, such destabilization will tend to occur (see [18] for a more detailed development), as will be illustrated in the example of Section 5.

4 Coupling Identification and Control

The poor conditioning of sheet and film processes makes it difficult to reliably identify the signs of all of the process gains $\Lambda_{t,ii}(0)$ from the limited input-output data usually available for sheet and film processes (this will be illustrated using rigorous statistics in the next section). This motivates the idea of designing the controller to only perform manipulations in directions that correspond to model gains $\Lambda_{m,ii}(0)$ whose signs are known with confidence (these directions are the corresponding columns V_i of the process input rotation matrix V). Manipulations in directions corresponding to process gains whose signs may be incorrectly identified may lead to poor closed loop performance. These considerations, and the proven robustness and optimality properties of the SVD controller (6), motivate the following coupling between the identification procedure and the controller design.

Identification The input-output data are collected and used to statistically fit a model for the process dynamics $g(s)$ (in (3)) and the steady state process interaction matrix $P_m(0)$. The scalar dynamics for sheet and film processes are well described as being first order plus time delay (the approach can be generalized), and are scaled so that $g(0) = 1$. The process model gains $\Lambda_{m,ii}(0)$ are computed by taking a matrix decomposition of the steady state interactions matrix $P_m(0) = U\Lambda_m(0)V^T$ estimated from the input-output data. This results in

$$P_m(s) = \frac{e^{-\theta s}}{\tau s + 1} P_m(0) = \frac{e^{-\theta s}}{\tau s + 1} U\Lambda_m(0)V^T = \frac{e^{-\theta s}}{\tau s + 1} \sum_{i=1}^{n} \Lambda_{m,ii}(0) U_i V_i^T. \tag{7}$$

Controller Design Basic statistical analysis is used to compute confidence intervals for each $\Lambda_{m,ii}(0)$ [9, 15]. The SVD controller is designed to only make manipulated variable moves in the directions V_i corresponding to gains $\Lambda_{m,ii}(0)$ that are known with confidence. The confidence intervals describe the accuracy of each controlled gain, and are used to design a controller robust to the potential gain variations. Inaccuracies in U and V can be accounted for by output and input uncertainty descriptions during the controller design.

The robust controller design procedure, described in detail elsewhere [22, 7] is summarized below. First, standard block diagram algebraic manipulations (either by hand [29, 35] or using off-the-shelf programs [2, 13, 31, 32]) are used to construct the standard uncertain system representation in Figure 2. The test for robust stability is that

$$\sup_{\omega} \mu(F_l(N(j\omega), K(j\omega))) < 1, \forall \omega, \tag{8}$$

where $F_l(N(j\omega), K(j\omega))$ is the lower Linear Fractional Transformation (LFT) between N and K and μ is the robustness margin (for background on robustness analysis see [16, 33, 29, 35]). It is important to note that the left hand side of (8) is computable from the controller K, the plant P, and the uncertainty description. In general, robustness margin computations are computationally expensive for processes of high dimension [8, 4]. However, with pseudo-SVD processes, the dimension of the problem can be reduced for a broad class of realistic uncertainty descriptions, including descriptions which capture inaccuracies in the process input and output rotation matrices V and U (for details see [7]). The robustness margin for pseudo-SVD processes can be computed as

$$\mu(F_l(N(j\omega), K(j\omega))) = \max_i \left\{ \mu(F_l(N^i(j\omega), \Lambda_{K,ii}(j\omega))) \right\}, \tag{9}$$

where N^i corresponds to a single-input single-output (SISO) robust controller design problem for each controlled gain $\Lambda_{m,ii}(s)$. This decouples the design problem for $K(s)$ into *independent* design problems for each $\Lambda_{K,ii}(s)$. The controller $\Lambda_{K,ii}(s)$ for each SISO problem can be designed by any robust controller method; we suggest to use IMC tuning [29]:

$$\Lambda_{K,ii}(s) = \begin{cases} \dfrac{1}{\Lambda_{m,ii}(0)} \cdot \dfrac{\left(1 + \tau_D s + \dfrac{1}{\tau_I s}\right)}{\tau_{F,i} s + 1} \cdot \dfrac{2\tau + \theta}{2(\lambda_i + \theta)} & \text{for } \Lambda_{m,ii} \text{ known with confidence} \\ 0 & \text{otherwise} \end{cases} \tag{10}$$

with

$$\tau_I = \tau + \frac{\theta}{2}; \quad \tau_D = \frac{\tau\theta}{2\tau + \theta}; \quad \tau_{F,i} = \frac{\lambda_i \theta}{2(\lambda_i + \theta)}.$$

The SISO controllers $\Lambda_{K,ii}(s)$ are stacked up as the diagonal elements of a matrix $\Lambda_K(s)$, with the overall SVD controller computed from

$$K(s) = V\Lambda_K(s)U^T. \tag{11}$$

	$c_1(0)$	$c_2(0)$	$c_3(0)$	$c_4(0)$	$c_5(0)$	$c_6(0)$	$c_7(0)$	$c_8(0)$
Model	1.2811	0.8425	0.6267	0.2155	0.0706	-0.2249	0.0252	0.0249
True	1.0	0.9	0.6	0.2	0.1	-0.1	0.05	0.0

Table 1: Comparison of the process parameters of the model to the true process for the blown film example.

The IMC tuning parameters λ_i are selected as fast as possible while maintaining stability for all model gain variations. Any of the well-established multivariable antiwindup procedures can be used to deal with the constraints [1, 11, 28, 40]. Constraint handling may be unnecessary for many sheet and film processes, since directions corresponding to low steady-state gains $\Lambda_{m,ii}(0)$ are not manipulated by the SVD controller, and designing the controller to be robust tends to prevent overly large dynamic excursions in the manipulated variables. This is illustrated in the following example.

5 Example

The following example illustrates the relationship between process identification and the controller design, when applied to a large scale blown film process.

The example to be considered is a blown film plastic extruder with 45 actuator and measurement locations. Since the actuator die is circular, the circular symmetric process description is applicable. The steady state interaction matrix was assumed to have 8 parameters. The process model was identified with 5 bump tests and the values of the c_j were calculated using least squares. The c_j for both the model and the true process are shown in Table 1.

Figure 3 shows the confidence interval associated with each (distinct) identified process gain, as well as the values of the gains for the true process. Based on the criteria that the interval must not include zero, eleven of the twenty-three distinct gains were not known with confidence. Of these eleven, five were actually identified incorrectly. The problem was then reduced to designing SISO controllers for the remaining twelve gain directions. Actuator and gain uncertainty descriptions with the robust controller design procedure were used, that is, each SISO controller was chosen to be IMC-PID, and the tuning parameter λ_i was optimized for each subproblem.

The full SVD controller was then tested against the QPF method at three different constraint levels (listed in Table 2) of the form

$$\begin{array}{ll} u_l \le u_i \le u_h, & \text{(min-max constraint)} \\ -l_b \le u_{i-1} - 2u_i + u_{i+1} \le l_b & \text{(2nd-order bending moment constraint)} \end{array} \tag{12}$$

for an initial disturbance profile d consisting of a summation of sinusoids

$$d_k = \sin\left(\frac{(2\pi)4k}{45}\right) + \sin\left(\frac{(2\pi)6k}{45}\right) + \sin\left(\frac{(2\pi)18k}{45}\right); \quad k = 1, \ldots, 45. \tag{13}$$

In Case 1 where the constraints are very tight, the QPF method performs better than the SVD controller (see Fig. 4). However, when the constraints are eased to Case 2, the QPF performs worse than the SVD (Fig. 5), and for Case 3 the QPF method is worse than open-loop (Fig. 6). The manipulated variable moves for the SVD controller (figure not shown for brevity) easily satisfy the tightest constraint set (Case 1), with a maximum deviation of 0.3744 and a maximum

	Case 1	Case 2	Case 3
l_b	1	5	10
$u_h, -u_l$	1	10	10

Table 2: The constraints used three simulations for the QPF method.

second-order bending moment of 0.2533. The weaker constraints are not unreasonable for a blown film process where the thickness is controlled by cooling jets, since there would be no bending constraint and the actuators would have a wider range of operation. The standard deviations of all the profiles are listed in Table 3.

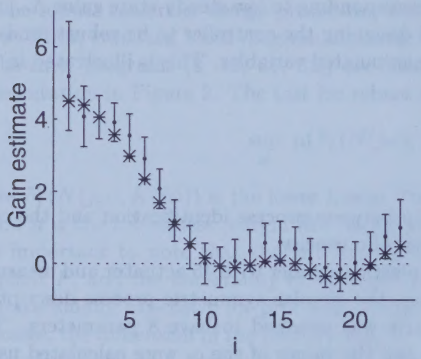


Figure 3: The plant gains $\Lambda_{t,ii}$ (shown using $*$) and the confidence intervals for each model gain $\Lambda_{m,ii}$.

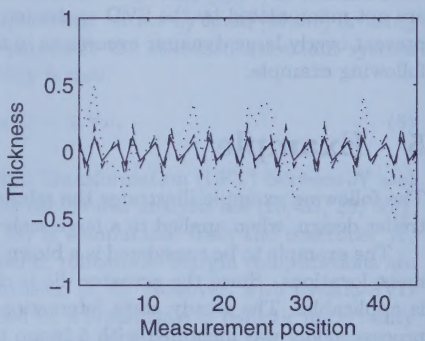


Figure 4: Steady state profile with QPF at Case 1 constraints where: QPF (—); SVD (– –); initial disturbance (· · ·).

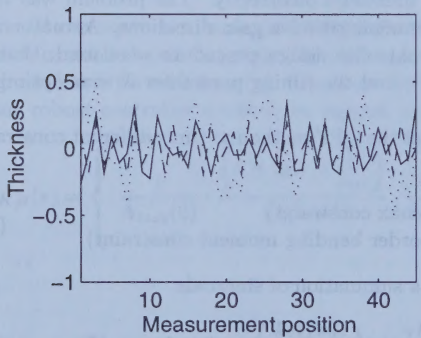


Figure 5: Steady state profile with QPF at Case 2 constraints where: QPF (—); SVD (– –); initial disturbance (· · ·).

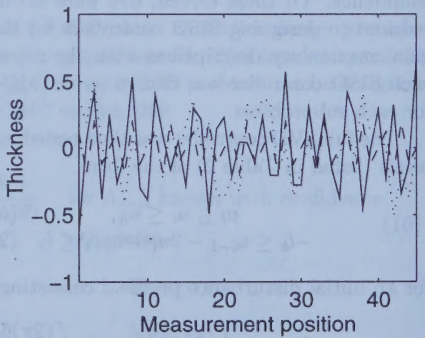


Figure 6: Steady state profile with QPF at Case 3 constraints where: QPF (—); SVD (– –); initial disturbance (· · ·).

	Initial Disturbance	SVD	QPF Case 1	QPF Case 2	QPF Case3
σ	0.2477	0.1430	0.0772	0.1618	0.3137

Table 3: The standard deviation of the initial disturbance profile and steady state profiles after control for the blown film example.

6 Conclusions

The interaction between model accuracy and closed loop performance was explored for sheet and film processes using a model decomposition in terms of a static input rotation matrix (V), a diagonal transfer function matrix ($\Lambda_m(s)$), and a static output rotation matrix (U). Theoretical results and simulations indicated the importance of only performing manipulations in directions of the process input vectors (V_i) corresponding to steady-state gains whose sign has been reliably identified. The SVD controller is designed to be robust to inaccuracies in the controlled model gains and in the input and output rotation matrices (the examples considered the case with potential inaccuracies in the input rotation matrix). The performance of the SVD controller was compared to that of the industrially-accepted QPF controller design method. While attempting to control in all directions was shown to result in poor performance, the SVD controller provided consistently good results. Furthermore, the simulation examples suggested that constraint handling may be unnecessary for many sheet and film processes when the SVD identification and control procedure is used. This is because directions corresponding to low steady-state gains $\Lambda_{m,ii}(0)$ are not manipulated by the SVD controller, and the designing of the SVD controller to be robust tends to prevent overly large dynamic excursions in the manipulated variables. In cases where constraint handling is necessary, any of the standard multivariable antiwindup methods proposed by Brosilow and others can be used. This results in a simple controller implementation.

Although the focus of this manuscript was on sheet and film processes, the results also have relevance to any process that can be represented in pseudo-SVD form, a comprehensive list of which appears in the first author's Masters thesis.

Acknowledgements

Support from the DuPont Young Faculty Award is gratefully acknowledged.

References

[1] K. J. Åström and B. Wittenmark. *Computer Controlled Systems: Theory and Design.* Prentice-Hall, Inc., Englewood Cliffs, N.J., 1984.

[2] G. J. Balas, J. C. Doyle, K. Glover, A. K. Packard, and R. S. R. Smith. *μ-Analysis and Synthesis Toolbox (μ-Tools) : Matlab Functions for the Analysis and Design of Robust Control Systems.* The MathWorks, Inc., Natick, Massachusetts, 1992.

[3] T. J. Boyle. Practical algorithms for cross-direction control. *Tappi*, 61:77–80, 1978.

[4] R. D. Braatz. Robustness margin computation for large scale systems. In *AIChE Annual Meeting*, Chicago, Illinois, 1996. to be presented.

[5] R. D. Braatz, B. A. Ogunnaike, and A. P. Featherstone. Identification, estimation, and control of sheet and film processes. In *Proc. of the IFAC World Congress*, pages 319–324, San Francisco, California, 1996. Elsevier Science Inc.

[6] R. D. Braatz, M. L. Tyler, M. Morari, F. R. Pranckh, and L. Sartor. Identification and cross-directional control of coating processes. *AIChE J.*, 38:1329–1339, 1992.

[7] R. D. Braatz and J. G. VanAntwerp. Robust cross-directional control of large scale paper machines. In *Proc. of the IEEE International Conf. on Control Applications*, pages 155–160, Dearborn, Michigan, September 15-18, 1996.

[8] R. D. Braatz, P. M. Young, J. C. Doyle, and M. Morari. Computational complexity of μ calculation. *IEEE Trans. on Auto. Control*, 39:1000–1002, 1994.

[9] H. D. Brunk. *An Introduction to Mathematical Statistics.* Xerox College Publishing, Lexington, Massachusetts, 2nd edition, 1965.

[10] J. Callari. New extrusion lines run tighter, leaner. *Plastics World*, pages 28–31, Jan. 1990.

[11] P. J. Campo and M. Morari. Robust control of processes subject to saturation nonlinearities. *Comp. & Chem. Eng.*, 14:343–358, 1990.

[12] P. J. Campo and M. Morari. Achievable closed loop properties of systems under decentralized control: Conditions involving the steady state gain. *IEEE Trans. on Auto. Control*, 39:932–943, 1994.

[13] R. Y. Chiang and M. G. Safonov. *Robust Control Toolbox: For Use with MATLAB.* The MathWorks, Inc., Natick, Massachusetts, 1992.

[14] W. H. Cuffey. Some factors involved in basis weight uniformity. *Tappi*, 40:190A–196A, 1957.

[15] J. L. Devore. *Probability and Statistics for Engineering and the Sciences.* Brooks/Cole Publishing, Monterey, California, 1982.

[16] J. C. Doyle. Analysis of feedback systems with structured uncertainties. *IEE Proc. Pt. D*, 129:242–250, 1982.

[17] A. P. Featherstone. *Control Relevant Identification of Structured Large Scale Systems*, M.S. thesis, University of Illinois, Urbana, Illinois, 1995.

[18] A. P. Featherstone. *Control Relevant Identification of Large Scale Sheet and Film Processes*, Ph.D. thesis, University of Illinois, Urbana, Illinois, 1996. in preparation.

[19] A. P. Featherstone and R. D. Braatz. Control relevant identification of sheet and film processes. In *Proc. of the American Control Conf.*, pages 2692–2696, Seattle, Washington, 1995.

[20] G. H. Golub and C. F. van Loan. *Matrix Computations.* Johns Hopkins University Press, Baltimore, Maryland, 1983.

[21] A. Halouskova, M. Karny, and I. Nagy. Adaptive cross-direction control of paper basis weight. *Automatica*, 29:425–429, 1993.

[22] M. Hovd, R. D. Braatz, and S. Skogestad. SVD controllers for H_2-, H_∞-, and μ-optimal control. In *Proc. of the American Control Conf.*, pages 1233–1237, Baltimore, Maryland, 1994.

[23] H. Karlsson and L. Haglund. Optimal cross-direction basis weight and moisture profile control on paper machines. In *Proc. of the 3rd Int. Pulp and Paper Process Control Symp.*, Vancouver, B. C., Canada, 1983.

[24] A. P. Kjaer, W. P. Heath, and P. E. Wellstead. Identification of cross-directional behaviour in web production: techniques and experience. In *Proc. of the IFAC Symp. on System Identification*, Copenhagen, Denmark, July 1994.

[25] K. Kristinsson and G. A. Dumont. Paper machine cross directional basis weight using Gram polynomials. *Automatica*, 32:533–548, 1996.

[26] D. Laughlin, M. Morari, and R. D. Braatz. Robust performance of cross-directional basis-weight control in paper machines. *Automatica*, 29:1395–1410, 1993.

[27] R. Martino. Motor-driven cams actuator flexible-lip automatic die. *Modern Plastics*, 68:23–23, 1991.

[28] S. Mhatre and C. B. Brosilow. Multivariable model state feedback. In *Proc. of the Second Midwest Process Control Workshop*, University of Michigan, Ann Arbor, Michigan, April 27, 1996. also in the *Proc. of the IFAC World Congress*, San Francisco, California, June 30 - July 5, 1996.

[29] M. Morari and E. Zafiriou. *Robust Process Control.* Prentice-Hall, Englewood Cliffs, New Jersey, 1989.

[30] G. A. Richards. Cross direction weight control. *Japan Pulp and Paper*, pages 41–53, Nov. 1982.

[31] E. L. Russell and R. D. Braatz. Analysis of large scale systems with model uncertainty, actuator and state constraints, and time delays. In *AIChE Annual Meeting*, Chicago, Illinois, Nov. 1996. to be presented.

[32] E. L. Russell, C. P. H. Power, and R. D. Braatz. Multidimensional realizations of large scale uncertain systems for multivariable stability margin computation. *Int. J. of Robust and Nonlinear Control*, 1996. in press.

[33] M. G. Safonov. Stability margins of diagonally perturbed multivariable feedback systems. *IEE Proc. Pt. D*, 129:251–256, 1982.

[34] S. Skogestad, M. Morari, and J. C. Doyle. Robust control of ill-conditioned plants: High purity distillation. *IEEE Trans. on Auto. Control*, 33:1092–1105, 1988.

[35] S. Skogestad and I. Postlethwaite. *Multivariable Feedback Control: Analysis and Design.* Wiley, New York, 1996.

[36] R. M. Tong. Automatic control of grammage profile on a paper machine. In *Proc. of the 3rd IFAC PRP Conf.*, pages 289–298, Brussels, Belgium, 1976.

[37] B. W. Wallace. Economic benefits offered by computerized profile control of weight, moisture, and caliper. *Tappi*, 64:79–83, 1981.

[38] R. G. Wilhelm, Jr. and M. Fjeld. Control algorithms for cross directional control. In *Proc. of the 5th IFAC PRP Conf.*, pages 163–174, Antwerp, Belgium, 1983.

[39] A. J. Wilkinson and A. Hering. A new control technique for cross machine control of basis weight. In *Preprints of the 5th IFAC PRP Conf.*, pages 151–155, Antwerp, Belgium, 1983.

[40] A. Zheng, M. V. Kothare, and M. Morari. Antiwindup design for internal model control. *Int. J. of Control*, 60:1015–1024, 1994.

Performance Assessment of Interacting Processes

Othman Taha *, Guy A. Dumont * & John B. Ball **
Emails: Otaha@ppc.ubc.ca, Guyd@ppc.ubc.ca & JohnB@mail.canfor.ca
***: Department of Electrical Engineering & Pulp and Paper Centre**
University of British Columbia
2385 East Mall, Vancouver, BC, V6T1Z4, Canada.
****: P.G pulp and paper mill, Process control department**
P.O Box 6000, Prince George, BC, V2N2K3, Canada.

Abstract:

This paper addresses the use of a control loop performance monitoring system (CLPM) in a paper mill, with applications to a paper machine headbox and a kraft pulp digester. Both systems are multivariable and display strong coupling. CLPM indicates if the desired loop performance is met. If not, it provides tools to determine the origin of performance degradation and suggests corrective actions. Results indicate that loop malfunction is often due to interaction, process perturbations or actuators nonlinearities instead of poorly tuned controllers.

Keywords: Loop performance monitoring, Interaction, Headbox, Kraft pulp Digester.

Introduction:

Unlike standard identification methods, control loop performance monitoring approach assesses the level of performance from closed loop operating data only. These methods have found some industrial applications [Dumont 1992] with encouraging results. Minimum variance performance can be estimated from an arbitrary sample of routine closed loop data and used as a benchmark. Several authors have recently proposed statistics which amount to the normalization of the closed loop output error variance with respect to the best achievable variance. The nearness of this ratio to one means that the performance is close to minimum variance control.

The procedure for estimating the minimal variance is in 3 steps: i- process time delay estimation, ii- closed-loop data time series modeling, iii- computation of minimal variance as:

$$\sigma_{mv}^2 = (1 + \phi_1^2 + \phi_2^2 + \ldots + \phi_d^2)\sigma_e^2 \qquad (1)$$

where σ_{mv}^2 is the minimal achievable variance, σ_e^2 is the white noise variance and ϕ_i^2's are the times series Markov coefficients. The performance index is then:

$$p = \frac{\sigma_y^2}{\sigma_{mv}^2} \qquad (2)$$

The index in (2) expresses the amount by which the actual output variance exceeds the minimal variance. Although this index gives an interesting measure of the actual performance, it has some shortcomings. The index will depend on the accuracy of the delay estimation. The process perturbations are supposed to be stochastic, stationary. This condition is rarely met in industrial applications. The system is assumed to be stable,

minimum phase system. An extension to unstable, NMP systems hs been developed by [Tyler and Morari 1995]. In this case, the location of unstable poles and zeros has to be known.

In practice there are cases where the minimum variance index (2) alone is misleading. We thus use complementary tools such as oscillation detection in control loop, detection of friction in control valves, evaluation of interaction between loops, using Relative Gain Analysis and identification of the static input-output relationship.

[Tyler and Morari 1995] proposed similar approach using likelihood methods [M. Basseville and I. Nikiforov 1993]. Performance is tested as constraints on the pulse closed loop response. They define the generalized likelihood ratio test (GLR). This involves solving a constrained as well as an unconstrained model identification problem.

When a model of the system is available, model based methods such as the one proposed by [J. Kendra and A. Cinar 1996] can be used. Effect of external disturbances can be differentiated from problems with controller tuning. These methods however need more expertise for the choice of the type and order of models. When the system is nonlinear, on-line estimation should be used.

Headbox system:

The purpose of the headbox on a paper machine is to distribute the water/fibre suspension on the wire as evenly and steadily as possible. In a pressurized headbox, the total pressure determines the outlet velocity at the slice. For a system like a headbox, the performance is a major requirement since it directly influences final paper quality. A MIMO model of the headbox is developed and its parameters adjusted on line to the real process. With this model, influence of setpoint changes and perturbations can be separated easily. This system is subject to disturbances in the inflow rate of the thin stock of the headbox and in the pressure of the air supplied. Air flow and/or stock flow control valve nonlinearities are also present. The CLPM will help for investigating performance obtained with different control strategies.

Digester system:

The process unit operation include both advanced and regulatory loops, process interactions, and both fast and slow process dynamics. A total of 34 loops in a 500t/d kamyr continuous kraft pulp digester were chosen for the study. These were divided into three groups of process related loops: stochastic type chip column movement, slow dynamics non linear chip cooking , and fast dynamics chip feed system. Control loops are grouped together when there is some significant chance that cycling in one member of the group could propagate to another member. Qualitative knowledge of the process is the best guide for selecting this grouping.

Conclusion:

CLPM is a valuable tools for engineers responsible for thousands of loops in a plant by providing useful information on process disturbances, dynamic interactions and actuators/sensors nonlinearities or poor sizing. CLPM could help in designing and evaluating control strategies.

References:

Due to a lack of space, references could not be included. For a list of references, please contact G. Dumont at the address above.

RISK-SENSITIVE IDENTIFICATION OF DYNAMIC SYSTEMS BY NEURAL NETWORKS

James Ting-Ho Lo
Department of Mathematics and Statistics
University of Maryland Baltimore County
Baltimore, MD 21228
e-mail: lo@math.umbc.edu

Abstract

Risk-sensitive criteria for system identification emphasize or de-emphasize large individual errors in an exponential manner and thereby induce risk-averting or risk-seeking identification performances. Identification by neural networks usually is not perfect and some errors are bound to be present. The availability of the risk-sensitive criteria allows us to obtain neural networks as system identifiers to better suit a large variety of applications.

This paper shows that under mild regularity conditions, general risk-sensitive identification of dynamic systems by neural networks can be done to any desired degree of accuracy in both the series-parallel and parallel formulations.

I. Introduction

Identification of a dynamic system from the measurements of its input and output processes is a fundamental problem in system control, signal processing, robotics, seismology, communication, economics, time-series prediction, etc. While much progress has been made on the identification of systems with linear structures [7, 6], very few results have been reported on nonlinear system identification until recently. The recent development on the use of neural networks for system identification [10, 11, 2, 13, 8] points to a most promising approach.

Essentially, three neural network paradigms have been employed. The first is the multilayer perceptrons with tapped delay lines (MLPWTDs). They are used as building blocks in both the series-parallel and the parallel formulations in [10, 11].

The second paradigm that has been employed for system identification [2, 13, 8] is the multilayer perceptrons with the neurons in each hidden layer interconnected through a unit time delay device, which were first suggested in [1]. We denote such networks by MLPWINs (multilayer perceptron with interconnected neurons). In [13], an example nonlinear system is identified by a series-parallel formulation using an MLPWIN. It is noted in [12] that the numerical test results are significantly superior to those achieved by MLPWTDs.

In [8], MLPs with free and teacher-forced output feedbacks, to be denoted by MLPWOFs, are proposed for system identification. Both the MLPWTD and the MLPWOF use a pure MLP (without output feedbacks or interconnects among the hidden neurons) as the principal component. The only difference between them is that the MLPWOF has output feedbacks instead of tapped delay lines. After proper training, the free feedbacks

of the MLPWOF carry the optimal information for minimizing its external output errors and are hence more efficient than the tapped delay lines for storing and processing the information.

In all these existing results on system identification by neural networks, the identification error criteria are the standard mean square errors of the neural identifiers' outputs. In [8], it is proven that both an MLPWIN and an MLPWOF exist that approximate almost any dynamic system to any degree of accuracy with respect to the mean square errors in the series-parallel identification formulation and, for a finite time period and under some more regularity conditions on the system, in the parallel formulation as well.

In this paper, we propose to use the identification error criteria that are based on the risk-sensitive functionals instead of the standard mean square errors to achieve robust identification of dynamic systems by neural networks.

The risk-sensitive (or exponential cost) functionals were first proposed by Jacobson [5, 14] for optimal stochastic linear control. Their relations with deterministic games and H_∞ criteria for robust linear control and estimation have attracted a great deal of attention in the past few years [3, 16, 15, 9, 4].

In a companion paper also presented at the 34th Annual Allerton Conference on Communication, Control, and Computing, it is proven that under mild conditions, a function can be approximated arbitrarily closely with respect to general risk-sensitive functionals by multilayer perceptrons. Based on these risk-sensitive approximation results, this paper establishes that an MLP, an MLPWIN and an MLPWOF exist that approximate almost any dynamic system to any degree of accuracy with respect to some identification criteria based on the risk-sensitive functionals in the series-parallel identification formulation and, for a finite time period and under some more regularity conditions on the system, in the parallel formulation as well.

To simplify our presentation, only single-input-single-output (SISO) dynamic systems and neural networks with a single hidden layer of neurons are treated in this paper. However, all the discussions and results can be easily generalized to multiple-input-multiple-output (MIMO) systems and neural networks with more than one hidden layer of neurons.

II. Risk-Sensitive Approximation of Functions

In a companion paper also presented to the 34th Annual Allerton Conference, the standard risk-sensitive functional [5] is generalized and two theorems on the approximation of functions by MLPs and RBFs with respect to these risk-sensitive functionals are given. The general risk-sensitive functional and a corollary to the theorem on the approximation by MLPs are given below.

Definition 1 *Let p be a positive real number, λ be a real number, and $S_p(K)$ be the set of real-valued functions f defined on a compact set K in R^n with probability measure $\mu(K) = 1$ such that the Lebesgue integral,*

$$E\left[\exp\left(\lambda |f|^p\right)\right] := \int \exp\left(\lambda |f(x)|^p\right) \mu(dx) < \infty.$$

The order-(λ, p) risk-sensitive functional $\rho_{\lambda,p}$ on $S_p(K)$ is defined to be

$$\rho_{\lambda,p}(f) := \frac{1}{\lambda} \ln \int \exp\left(\lambda |f(x)|^p\right) \mu(dx).$$

Some intuition can be obtained by expanding $\rho_{\lambda,p}(f)$ in terms of λ and writing

$$\rho_{\lambda,p}(f) = E\left(|f|^p\right) + \frac{\lambda}{2} E\left(|f|^{2p}\right) + o\left(\lambda^2\right),$$

which shows that as $\lambda \to 0$, $\rho_{\lambda,p}(f)$ approaches $E(|f|^p)$. We define $\rho_{0,p}(f)$ to be the functional $E(|f|^p)$.

Theorem 1 *Let a be a continuous or nondecreasing sigmoidal function, and let K, μ, λ, p, $\rho_{\lambda,p}$, and E be defined as in Definition 1. Let $f : R^n \longrightarrow R^1$ be a Borel measurable function satisfying*

$$|f(x)| < B$$

for a positive number B and for all x in R^n. Then given any $\varepsilon > 0$, there are an integer N, real constants c_i, b_i, $i = 1, \ldots, N$, and row n-vectors $w_i = [w_{i1}, \ldots, w_{in}]$, $i = 1, \ldots, N$, such that

$$\rho_{\lambda,p}(f - s) < \varepsilon \tag{II.1}$$

$$s(x) := \sum_{i=1}^{N} c_i a(w_i.x + b_i). \tag{II.2}$$

Throughout the rest of this paper, the activation function a of every neuron is assumed to be a strictly increasing sigmoidal function.

III. Formulation of Risk-Sensitive Identification of Systems

Consider the discrete-time SISO causal time-invariant plant

$$y(t+1) = f(y(t), y(t-1), \ldots, y(t-p+1), u(t), u(t-1), \ldots, u(t-q+1)), \tag{III.1}$$

where $(u(t), y(t))$ is the input/output pair at time t. The finite integers p and q are unknown. It is assumed that for all t, $|u(t)| < B$, $|y(t)| < B$ and $f(B^{p+q}) \subset (-B, B)$ for some known $B < \infty$ and that $p < b$ and $q < b$ for some known $b < \infty$. No mathematical description for the function f is available. Nevertheless, it is well defined for the specification of the plant by all the possible sequences of the input and output pairs $(u(t), y(t))$.

The problem of identifying the plant (III.1) is to construct a mathematical system, called a system identifier, whose output $\hat{y}(t+1)$ at time t approximates that of the plant as closely as possible during the operations of the plant, using either the operational data or the experimental data or both. The data, whether operational or experimental, must be sufficiently representative of all the possible operational sequences of input/output pairs and sufficiently reflective of the joint probability distributions of u and y viewed as stochastic processes. We assume that both u and y are stationary stochastic processes.

For risk-sensitive identication of the plant (III.1), the order-(λ, p) risk-sensitive criterion is

$$\frac{1}{\lambda} \ln E\left[\exp \lambda \sum_{t=b+1}^{T} |y(t) - \hat{y}(t)|^p\right], \tag{III.2}$$

where T is a positive integer selected such that the operations of the plant in the time interval $(b+1, T)$ represent those of the plant over its typical continuous runs and $\hat{y}(t+1)$ denotes the output of the system identifier at time t. This criterion is similar to the control and estimation criteria used in [5, 15]. An alternative order-(λ, p) risk-sensitive criterion is

$$\frac{1}{\lambda} \ln E\left[\frac{1}{T-b} \sum_{t=b+1}^{T} \exp \lambda |y(t) - \hat{y}(t)|^p\right]. \tag{III.3}$$

The theorems to be stated and proven in the sequel are valid with the criterion (III.2) replace with the criterion (refrscriter1). It is easy to see how the given proofs are modified to suit (refrscriter1).

There are essentially two formulations for dynamic system identification, namely the series-parallel formulation and the parallel formulation. In the series-parallel formulation, the system identifier inputs $u(t)$ and $y(t)$ at time t. In the parallel formulation, the system identifier inputs $u(t)$ and $\hat{y}(t)$ at time t.

IV. Series-Parallel Identification by ANNs

Theorem 2 *Consider the series-parallel identification of the plant (III.1). Given $\varepsilon > 0$, there are an MLPWTD, MLPWOF, and MLPWIN that input $u(t-1)$ and $y(t-1)$ and outputs $\alpha(t)$ at time t such that*

$$\frac{1}{\lambda} \ln E \left[\exp \lambda \sum_{t=b+1}^{T} |y(t) - \alpha(t)|^p \right] < \varepsilon, \tag{IV.1}$$

for any given real number λ and positive real number p.

Proof. We will first prove the existence of an MLPWTD such that (IV.1) holds. The proof will then be extended to the existence of an MLPWOF and MLPWIN.
Case 1: $\lambda > 0$. Consider the function $f_0(\xi) := f(\eta)$, where $\xi := [\xi_1, \ldots, \xi_{2b}]$ and $\eta := [\xi_1, \ldots, \xi_r, \xi_{b+1}, \ldots, \xi_{b+q}]^T$. Note that by this definition of f_0 and the boundedness assumption on f, we have $\left| f_0\left(B^{2b}\right) \right| = \left| f\left(B^{r+q}\right) \right| \subset (-B, B)$. By Theorem 1, given $\varepsilon > 0$, an MLP, which is a function $g : B^{r+q} \to R$, can be selected such that the error $\rho_{\lambda(T-b),p}(f_0 - g)$, where the probability measure μ is defined by $y(t), \ldots, y(t-r+1), u(t), \ldots, u(t-q+1)$, is less than $\varepsilon/(T-b)$ at each time $t+1$.

Let two tapped delay lines hold $u(t), \ldots, u(t-b+1)$ and $y(t), \ldots, y(t-b+1)$ at time $t+1$ and let the selected MLP be used to transform them into $\alpha(t+1) = g(y(t), \ldots, y(t-b+1), u(t), \ldots, u(t-b+1))$. By a generalized Hölder inequality, the risk-sensitive identification criterion (III.2) satisfies

$$\begin{aligned}
& \frac{1}{\lambda} \ln E \exp \lambda \sum_{t=b+1}^{T} |y(t) - \alpha(t)|^p \\
\leq\ & \frac{1}{\lambda} \ln \Pi_{t=b+1}^{T} \left[E \exp \lambda (T-b) |y(t) - \alpha(t)|^p \right]^{1/(T-b)} \\
=\ & \frac{1}{\lambda(T-b)} \sum_{t=b+1}^{T} \ln E \exp \lambda (T-b) |y(t) - \alpha(t)|^p \\
=\ & (T-b) \rho_{\lambda(T-b),p}(f_0 - g) < \varepsilon.
\end{aligned}$$

Case 2: $\lambda \leq 0$. Let f_0 be defined as before. Given $\varepsilon > 0$, an MLP, which is a function $g : B^{r+q} \to R$, can be selected such that the error $E|f_0 - g|^p \; ¡ \; 1/(T-b)$, where the probability measure μ is defined by $y(t), \ldots, y(t-r+1), u(t), \ldots, u(t-q+1)$, is less than $\varepsilon/(T-b)$ at each time $t+1$. Then the risk-sensitive identification criterion (III.2) satisfies

$$\frac{1}{\lambda} \ln E \exp \lambda \sum_{t=b+1}^{T} |y(t) - \alpha(t)|^p$$

$$\leq \frac{1}{\lambda} \ln \exp \lambda \sum_{t=b+1}^{T} E\,|y(t) - \alpha(t)|^p$$

$$= \sum_{t=b+1}^{T} E\,|f_0 - g|^p < \varepsilon.$$

This completes the proof for $\lambda \leq 0$.

We will now extend the above proof to the existence of an MLPWOF and MLPWIN. Connect $2b-2$ hidden neurons in series in such a way that these neurons and the input terminals of the ANN (MLPWOF or MLPWIN) form two tapped delay lines for holding $u(t), \ldots, u(t-b+1)$ and $y(t), \ldots, y(t-b+1)$ at time $t+1$. The rest of the proof is identical to the proof above for the existence of an MLPWTD.

V. Parallel Identification by ANNs

Theorem 3 *Consider the plant (III.1) with the range of f contained in $(-B, B)$ for some $B < \infty$ and with the $L_{p'}$ norm $\| \nabla_y f(y(t), \ldots, y(t-r+1), u(t), \ldots, u(t-q+1))\|_{p'}$ of the gradient $\nabla_y f$ with respect to $(y(t), \ldots, y(t-r+1))$ uniformly bounded by M, where $p' = p/(p-1)$. Assume that for all t, $|u(t)| < B$ and $|y(t)| < B$ and that the finite integers r and q are less than or equal to $b < \infty$. Given $\varepsilon > 0$, there exist an appropriately initialized MLPWTD, MLPWOF and MLPWIN, all with a single hidden layer of neurons, such that*

$$\frac{1}{\lambda} \ln E \left[\exp \lambda \sum_{t=b+1}^{T} |y(t) - \alpha(t)|^p \right] < \varepsilon, \tag{V.1}$$

where $\alpha(t)$ denotes the output of the MLPWOF at time t which has taken $\{(u(\tau), \alpha(\tau)), \tau = b+1, \ldots, t-1\}$ as the inputs in the given order.

Proof. We will first prove the existence of an MLPWOF. How the proof is extended for the existence of an MLPWIN will then be discussed. The extention of the proof to an MLPWTD is trivial and thus omitted.

Let an MLPWOF with $J := r + q - 2$ free output feedbacks, $\beta_1(t), \ldots, \beta_J(t)$ be initialized as follows: At time $t = 0$, set $\beta_1(t+1) = \cdots = \beta_J(t+1) = B$ and $\alpha(t+1) = y(1)$ at the output terminals for feedbacks. At times $t = 1, \ldots, b-1$, set $\alpha(t+1) = y(t+1)$ at the teacher-forced output terminal and $u(t) = u(t)$ at the external input terminal. Here the first $u(t)$ denotes the input terminal and the second $u(t)$ denotes the input at time t. At this point, the MLPWOF is ready to run by itself, starting with taking the input $u(b)$ and producing the output $\alpha(b+1)$.

Construct $r-1$ free feedback loops so that for $t = b, \ldots, T-1$ and $\tau = 2, \ldots, r-1$,

$$\beta_\tau(t) = a\,(\beta_{\tau-1}(t-1)), \tag{V.2}$$

and

$$\beta_1(t) = a(\alpha(t)). \tag{V.3}$$

Construct $q-1$ additional free feedback loops so that for $t = b, \ldots, T-1$ and $\tau = r+1, \ldots, r+q-2$,

$$\beta_\tau(t) = a\,(\beta_{\tau-1}(t-1)), \tag{V.4}$$

and

$$\beta_r(t) = a(u(t)). \tag{V.5}$$

At time $t+1$, the input vector to the MLPWOF is

$$\begin{aligned} z(t) \;:=\; & [\alpha(t), a(\alpha(t-1)), a^{\circ 2}((\alpha(t-2)), \ldots, a^{\circ(r-1)}(\alpha(t-r+1)), \\ & u(t), a(u(t-1)), a^{\circ 2}(u(t-2)), \ldots, a^{\circ(q-1)}(u(t-q+1))]^T, \end{aligned} \tag{V.6}$$

where $a^{\circ\tau}$ denotes the composition $a \circ a \circ \ldots \circ a$ of τ copies of a. Notice that $z(t)$ can be viewed as a random vector whose range is contained in the compact set $K := [-B, B] \times [-1, 1]^{r-1} \times [-B, B] \times [-1, 1]^{q-1}$.

The input terminals receiving the input vector (V.6), all the hidden neurons except those used above to construct the free feedback loops, and the external output terminals form an MLP inside the MLPWOF. Denoting the function representing the MLP by g, the external output of the MLPWOF at time $t+1$ is

$$\begin{aligned} \alpha(t+1) \;=\; & g(\alpha(t), a(\alpha(t-1)), \ldots, a^{\circ(r-1)}(\alpha(t-r+1)), \\ & u(t), a(u(t-1)), \ldots, a^{\circ(q-1)}(u(t-q+1))). \end{aligned}$$

In the following, we shall first establish an inequality (V.7). The following symbols will be needed.

$$\begin{aligned} u^t &:= (u(t), u(t-1), \ldots, u(t-q+1)) \\ \alpha^t &:= (\alpha(t), \alpha(t-1), \ldots, \alpha(t-r+1)) \\ y^t &:= (y(t), y(t-1), \ldots, y(t-r+1)) \\ f(y^t, u^t) &:= f(y(t), \ldots, y(t-r+1), u(t), \ldots, u(t-q+1)) \\ p(\alpha^t, u^t) &:= g(\alpha(t), a(\alpha(t-1)), \ldots, a^{\circ(r-1)}(\alpha(t-r+1)), \\ &\quad\; u(t), a(u(t-1)), \ldots, a^{\circ(q-1)}(u(t-q+1))) \\ c_p &:= 1, \text{ for } 0 < p \leq 1 \text{ and } := 2^{p-1}, \text{ for } 1 < p \\ d_t &:= \alpha(t) - y(t) \\ \delta^t &:= p(\alpha^t, u^t) - f(\alpha^t, u^t) \\ a_i &:= c_p (c_p M^p)^{i-1} \sum_{t=b-i+1}^{T-i} \sum_{\tau_1=t-b+1}^{t} \cdots \sum_{\tau_{i-1}=\tau_{i-2}-b+1}^{\tau_{i-2}} |\delta^{\tau_{i-1}}|^p \end{aligned}$$

Note that $y(t+1) = f(u^t, y^t)$ and $\alpha(t+1) = p(u^t, \alpha^t)$.

By algebraic inequalities, the Hölder inequality, a mean value theorem and the given $L_{p'}$ bound M of the gradient $\nabla_y f$, we have

$$\begin{aligned} |d_{t+1}|^p &\leq \left[\left|f(y^t, \alpha^t) - f(y^t, u^t)\right| + \left|\delta^t\right|\right]^p \\ &\leq c_p \left|f(y^t, \alpha^t) - f(y^t, u^t)\right|^p + a_1 \\ &\leq c_p \left|\nabla_y f\left(y^t + \nu\left(\alpha^t - y^t\right), u^t\right)\left(\alpha^t - y^t\right)\right|^p + c_p \left|\delta^t\right|^p \\ &\leq c_p \left\|\nabla_y f\left(y^t + \nu\left(\alpha^t - y^t\right), u^t\right)\right\|_{p'}^p \left\|\alpha^t - y^t\right\|_p^p + c_p \left|\delta^t\right|^p \\ &\leq c_p M^p \sum_{\tau=t-b+1}^{t} |d_\tau|^p + c_p \left|\delta^t\right|^p. \end{aligned}$$

Applying this inequality repeatedly and noting that $d_t = 0$ for $t \leq b$ yield

$$\sum_{t=b+1}^{T} |d_t|^p \;\leq\; c_p M^p \sum_{t=b}^{T-1} \sum_{\tau_1=t-b+1}^{t} |d_{\tau_1}|^p + a_1$$

$$
\begin{aligned}
&\leq (c_p M^p)^2 \sum_{t=b-1}^{T-2} \sum_{\tau_1=t-b+1}^{t} \sum_{\tau_2=\tau_1-b+1}^{\tau_1} |d_{\tau_2}|^p + \sum_{i=1}^{2} a_i \\
&\quad \cdots \\
&\leq (c_p M^p)^{T-b-1} \sum_{t=b-(T-b)}^{T-(T-b-1)} \sum_{\tau_1=t-b+1}^{t} \cdots \sum_{\tau_{T-b-1}=\tau_{T-b-2}-b+1}^{\tau_{T-b-2}} |d_{\tau_{T-b-1}}|^p \\
&\quad + \sum_{i=1}^{T-b-1} a_i \\
&= (c_p M^p)^{T-b-1} |d_{b+1}|^p + \sum_{i=1}^{T-b-1} a_i \\
&\leq c_p (c_p M^p)^{T-b-1} |\delta^b|^p + \sum_{i=1}^{T-b-1} a_i.
\end{aligned}
$$

Notice that

$$
\begin{aligned}
\sum_{i=1}^{T-b-1} a_i &= c_p \sum_{i=1}^{T-b-1} \sum_{t=b+1}^{T-i} (c_p M^p b)^{i-1} |\delta^t|^p \\
&= c_p \sum_{t=b+1}^{T-1} \left[\sum_{i=1}^{T-t} (c_p M^p b)^{i-1} \right] |\delta^t|^p \\
&= c_p \sum_{t=b+1}^{T-1} \left[(c_p M^p b)^{T-t} - 1 \right] |\delta^t|^p / (c_p M^p b - 1).
\end{aligned}
$$

Denoting $\max \left\{ \left[c_p (c_p M^p b)^{T-t} - 1 \right] / (c_p M^p b - 1), c_p (c_p M^p)^{T-b-1} \right\}$ by K, we obtain the desired inequality

$$
\sum_{t=b+1}^{T} |d_t|^p \leq K \sum_{t=b}^{T-1} |\delta^t|^p. \tag{V.7}
$$

Case: $\lambda > 0$. By a generalized Hölder inequality, it follows from (V.7) that

$$
\begin{aligned}
\frac{1}{\lambda} \ln E \exp \lambda \sum_{t=b+1}^{T} |d_t|^p &\leq \frac{1}{\lambda (T-b)} \sum_{t=b}^{T-1} \ln E \exp (T-1) \lambda K |\delta^t|^p \\
&= \frac{1}{\lambda} \ln E \exp (T-1) \lambda K |\delta^t|^p.
\end{aligned} \tag{V.8}
$$

By Theorem 1, given any $\varepsilon > 0$, there is an MLP represented by the function g such that

$$
\frac{1}{\lambda} \ln E \exp (T-1) \lambda K |\delta^t|^p < \varepsilon. \tag{V.9}
$$

Combining (V.8) and (V.9) completes the proof for $\lambda > 0$.

Case 2: $\lambda \leq 0$. By the well-known L_p function approximation capability of the MLPs, given any $\varepsilon > 0$, there is an represented by the function g such that $E |\delta^t|^p < \varepsilon / (T-b) K$. By (V.7) and Jensen's inequality,

$$
\frac{1}{\lambda} \ln E \exp \lambda \sum_{t=b+1}^{T} |d_t|^p \leq \frac{1}{\lambda} \ln E \exp \lambda K \sum_{t=b}^{T-1} |\delta^t|^p
$$

$$\leq \frac{1}{\lambda} \ln \exp \lambda K \sum_{t=b}^{T-1} E \left| \delta^t \right|^p$$

$$\leq \varepsilon,$$

completing the proof for $\lambda \leq 0$.

We will now consider the proof of the existence of an MLPWIN such that (V.1) holds. An MLPWIN has only one input terminal reserved for the exogenous input $u(t)$. Then how do we initialize the MLPWIN to incorporate $u(b-1), \ldots, u(1), y(b), \ldots, y(1)$ and then feedback the external output $\alpha(t)$ into the network? One way is to designate a hidden neuron, say neuron 1, for these tasks, which we call the initialization neuron. The same weights for the output terminal $\alpha(t+1)$ are assigned to the interconnects leading into the initialization neuron. Thus the activation level $\beta_1(t+1)$ of the initialization neuron is equal to $a(\alpha(t+1))$. After a unit time delay, it (i.e. $a(\alpha(t))$) is distributed as if it were an input terminal. Since a is strictly monotone increasing, the feedback $a(\alpha(t))$ carries as much information as $\alpha(t)$.

We use $r-1$ hidden neurons to construct a feedback loop for holding $a^{\circ(i+1)}(\alpha(t-i))$ for $i = 1, \ldots, r-1$, and $q-1$ hidden neurons to construct a feedback loop for holding $a^{\circ(i+1)}(u(t-i))$ for $i = 1, \ldots, q-1$. These feedback loops are similar to the two free feedbackloops constructed within the foregoing MLPWOF.

The MLPWIN is initialized as follows: At time $t = 0$, set $\beta_1(t+1) = a(y(t+1))$ and set the activation levels of all the other neurons in the MLPWIN, $\beta_j(t+1) = B$ for $j = 2, \ldots, r$, and $t = 1, \ldots, r$. At times $t = 1, \ldots, b-1$, set $\beta_1(t+1) = a(y(t+1))$ at the initialization neuron and $u(t) = u(t)$ at the external (the only) input terminal. At time $t = b$, the MLPWIN is ready to run by itself by starting with taking the first input $u(b)$ and producing the first output $\alpha(b+1)$.

From this point onward, the proof for the existence of an MLPWIN such that (V.1) holds is almost verbatim the same as the above proof for the existence of an MLPWOF.

VI. Conclusion

It has been established that under mild conditions, a dynamic system can be identified in the series-parallel formulation, to any desired degree of accuracy with respect to the general risk-sensitive criteria, by an MLPWTD, an MLPWOF, or an MLPWIN. It has also been established that for a finite time period, under some additional conditions, a dynamic system can be identified in the parallel formulation, to any desired degree of accuracy with respect to the general risk-sensitive criteria, by an MLPWTD, an MLPWOF or an MLPWIN as well.

Risk-sensitive and/or adaptive filtering and control by neural networks have been developed by the author. Results will soon be reported in forthcoming papers.

VII. REFERENCES

[1] J. L. Elman. Finding structure in time. Technical Report CRL 8801, Center for Research in Language, University of California, San Diego, 1988.

[2] B. Fernandez, A. G. Parlos, and W. K. Tsai. Nonlinear dynamic system identification using artifitial neural networks (ANNs). In *Proceedings of IJCNN-90*, pages 133–141, San Diego, CA, 1990.

[3] K. Glover and J. C. Doyle. State-space formulae for all stabilizing controllers that satisfy an h-infinity norm bound and relations to risk-sensitivity. *Systems Control Letters*, 11:167–172, 1988.

[4] B. Hassibi, A. H. Sayed, and T. Kailath. $h^{i}nfty$ optimality of the LMS algorithm. *IEEE Transactions on Signal Processing*, 44:267–280, 1996.

[5] D. H. Jacobson. Optimal stochastic linear systems with exponential performance criteria and their relation to deterministic games. *IEEE Transactions on Automatic Control*, AC-18-2:124–131, 1973.

[6] L. Ljung. *System Identification: Theory for the User.* Prentice Hall, Inc., Englewood Cliffs, 1987.

[7] L. Ljung and T. Soderstrom. *Theory and Practice of Recursive Identification.* MIT Press, Cambrdge and London, 1983.

[8] J. T. Lo. Dynamical system identification by recurrent multilayer perceptron. In *Proceedings of the 1993 World Congress on Neural Netwoks*, Portland, Oregon, July 1993.

[9] R. S. Mangoubi. Robust estimation and failure detection for linear systems. Technical Report Ph.D. Dissertation, Department of Aeronautics and Astronautics, MIT, Cambridge, MA, 1995.

[10] K. S. Narendra and K. Parthasarathy. Identification and control of dynamical systems using neural networks. *IEEE Transactions on Neural Networks*, 1(1):4–27, 1990.

[11] K. S. Narendra and K. Parthasarathy. Gradient method for the optimization of dynamical systems containing neural networks. *IEEE Transactions on Neural Networks*, 2(2):252–262, 1991.

[12] G. V. Puskorious and L. A. Feldkamp. Decoupled extended Kalman filter training of feedforward layered networks. In *Proceedings of IJCNN-91*, pages 771–777, Seatle, WA, 1991.

[13] G. V. Puskorious and L. A. Feldkamp. Model reference adaptive control with recurrent networks trained by the dynamic DEKF algorithm. In *Proceedings of IJCNN-92*, pages II106–113, Baltimore, MD, 1992.

[14] J. Speyer, J. Deyst, and D. H. Jacobson. Optimaization of stochastic linear systems with additive measurement and process noise using exponential performance criteria. *IEEE Transactions on Automatic Control*, AC-19:358–366, 1974.

[15] J. L. Speyer, C.-H. Fan, and R. N. Banavar. Optimal stochastic estimation with exponential cost criteria. In *Proceedings of the 31st Conference on Decision and Control*, New York, New York, 1992. IEEE.

[16] P. Whittle. *Risk Sensitive Optimal Control.* Wiley, New York, New York, 1990.

RISK-SENSITIVE APPROXIMATION OF FUNCTIONS BY NEURAL NETWORKS

James Ting-Ho Lo
Department of Mathematics and Statistics
University of Maryland Baltimore County
Baltimore, MD 21228
e-mail: lo@math.umbc.edu

Abstract

The standard risk-sensitive (or exponential quadratic) functional used for robust control and filtering for linear systems is generalized. It is then shown that under relatively mild conditions, a function can be approximated, to any desired degree of accuracy with respect to these general risk-sensitive functionals, by a multilayer perceptron or a radial basis function network.

I. Introduction

It is known that under mild conditions, functions can be approximated as closely as desired by artificial neural networks (ANNs) with respect to L_p norms [1, 2, 8, 12, 10, 13, 4]. This paper establishes the same with respect to the so-called risk-sensitive (or exponential cost) functionals (i.e. mappings from functions to real numbers).

The risk-sensitive functionals were first proposed by Jacobson [9, 14] for optimal stochastic linear control. Control and estimation criteria based on risk-sensitive functionals are known to be closely related with deterministic games and H_∞ criteria for robust linear control and estimation. In fact, such relations have attracted a great deal of attention in the past few years [3, 16, 15, 11, 5]. The standard risk-sensitive criteria emphasize greater errors in an exponential manner and thereby induce robust performances.

In this paper, we first generalize the standard risk-sensitive functional and observe that these risk-sensitive functionals except the risk-neutral ones (i.e. L_p norms) do not qualify as a norm for a space of functions. In fact, they do not satisfy either homogeneity or triangle inequality required of a norm. Nevertheless, it is then stated and proven that under mild conditions, a function can be approximated to any desired degree of accuracy with respect to any of these general risk-sensitive functionals by a multilayer perceptron (MLP) with as few as one hidden layer of neurons, whose activation functions are all either a continuous or a nondecreasing sigmoidal function.

The proof of this theorem is mainly based on the known fact that any continuous function can be approximated uniformly to an arbitrary accuracy by an MLP of the same type [1, 2, 8]. It is also known that many very widely used radial basis functions (RBFs) such as the Gaussian RBFs can also approximate any continuous function to an arbitray accuracy [12, 10, 13, 4]. This fact enables us to extend the foregoing theorem to function approximation by such RBFs. The proof of the extension is almost verbatim

the same as that of the foregoing theorem. Hence the extended theorem for those widely used RBFs is stated here without proof.

Further extensions of the foregoing theorems to other neural networks such as the high order or $\sum \Pi$ networks, that can uniformly approximate continous functions arbitrarily closely, can also be easily made. Due to the page limit, such further extensions are only briefly mentioned here.

To simplify our presentation, only the approximations of the multiple-input-single-output (MISO) functions by neural networks with a single hidden layer of neurons are treated in this paper. However, all the discussions and results can be easily generalized to the approximations of multiple-input-multiple-output (MIMO) functions by neural networks with more than one hidden layer of neurons.

II. Preliminaries

Definition 1 *A function* $a : R^1 \to R^1$ *is said to be sigmoidal, if*

$$a(x) \longrightarrow \begin{cases} 1 & as\ x \to +\infty, \\ -1 & as\ x \to -\infty. \end{cases}$$

The following theorem on the uniform approximation of continuous functions by MLPs is a combination of two theorems proven in [1, 2, 8].

Theorem 1 *Let* a *be a continuous or nondecreasing sigmoidal function, and let* K *denote a compact set in* R^n. *Given any continuous function* $f : R^n \to R^1$ *and an arbitrary* $\varepsilon > 0$, *there are an integer* N, *real constants* c_i, w_{ij}, b_i, $i = 1, \ldots, N$, $j = 1, \ldots, n$, *such that*

$$\left| f(x_1, \ldots, x_n) - \sum_{i=1}^{N} c_i a\left(\Sigma_{j=1}^{n} w_{ij} x_j + b_j\right) \right| < \varepsilon,$$

for all $(x_1, \ldots, x_n) \in K$.

The following theorem on the uniform approximation of continuous functions by RBFs is an immediate consequence of the classic Stone-Weierstrass theorem or Wiener approximation theorem [12, 10, 13, 4].

Theorem 2 *Let* $a(r, w_0)$ *be the Gaussian function* $\exp(-r^2/w_0^2)$, *the thin-plate-spline function* $r^2 \ln r$, *the multiquadric function* $(r^2 + w_0^2)^{1/2}$, *or the inverse multiquadric function* $(r^2 + w_0^2)^{-1/2}$, *where* w_0 *is a parameter and* r *is a real variable. Let* K *denote a compact set in* R^n. *Given any continuous function* $f : R^n \to R^1$ *and an arbitrary* $\varepsilon > 0$, *there are an integer* N, *real constants* w_0, c_i, w_{ij}, $i = 1, \ldots, N$, $j = 1, \ldots, n$, *such that*

$$\left| f(x_1, \ldots, x_n) - \sum_{i=1}^{N} c_i a\left(\sqrt{\Sigma_{j=1}^{n} (x_j - w_{ij})^2}, w_0\right) \right| < \varepsilon,$$

for all $(x_1, \ldots, x_n) \in K$.

III. General Risk-Sensitive Functionals

Definition 2 *Let p be a positive number and λ be a real number, and $S_p(K)$ be the set of real-valued functions f defined on a compact set K in R^n with probability measure $\mu(K) = 1$ such that the Lebesgue integral,*

$$E[\exp(\lambda |f|^p)] := \int \exp(\lambda |f(x)|^p)\, \mu(dx) < \infty.$$

The order-(λ, p) risk-sensitive functional $\rho_{\lambda,p}$ on $S_p(K)$ is defined to be

$$\rho_{\lambda,p}(f) := \frac{1}{\lambda} \ln \int \exp(\lambda |f(x)|^p)\, \mu(dx).$$

Some intuition can be obtained by expanding $\rho_{\lambda,p}(f)$ in terms of λ and writing

$$\rho_{\lambda,p}(f) = E(|f|^p) + \frac{\lambda}{2} E(|f|^{2p}) + o(\lambda^2),$$

which shows that as $\lambda \to 0$, $\rho_{\lambda,p}(f)$ approaches $E(|f|^p)$. We define $\rho_{0,p}(f)$ to be the functional $E(|f|^p)$.

The order-$(\lambda, 2)$ risk-sensitive functional for $\lambda > 0$ was proposed by D. Jacobson [9] and is called the standard risk-sensitive functional. We note that by the convexity of $(\cdot)^a$ for $a > 1$ (or the concavity of $(\cdot)^a$ for $a < 0$), Jensen's inequality, and the monotonicity of $\ln(\cdot)$, for $\lambda > 0$ and $c > 1$ (or for $\lambda < 0$ and $0 < c < 1$),

$$\begin{aligned} \frac{1}{\lambda} \ln E[\exp \lambda |cf|^p] &= \frac{1}{\lambda} \ln E\left[(\exp \lambda |f|^p)^{c^p}\right] \\ &\geq \frac{1}{\lambda} \ln (E[\exp \lambda |f|^p])^{c^p} \\ &= \frac{c^p}{\lambda} \ln (E \exp \lambda |f|^p) \end{aligned}$$

whence for $\lambda > 0$ and $c > 1$ (or for $\lambda < 0$ and $0 < c < 1$),

$$\begin{aligned} \rho_{\lambda,p}(cf) &\geq c^p \rho_{\lambda,p}(f) \\ [\rho_{\lambda,p}(cf)]^{\frac{1}{p}} &\geq c[\rho_{\lambda,p}(f)]^{\frac{1}{p}}, \end{aligned}$$

where the equalities hold if and only if f is a constant with probability one. This shows that neither $\rho_{\lambda,p}$ nor $\rho_{\lambda,p}^{1/p}$ satisfies homogeneity for $\lambda > 0$ and $c > 1$ (or for $\lambda < 0$ and $0 < c < 1$), and that neither $\rho_{\lambda,p}$ nor $\rho_{\lambda,p}^{1/p}$ satisfies triangle inequality for $\lambda > 0$ and $c > 1$. Note that homogeneity and triangle inequality are properties required of a norm.

Theorem 3 *If $E[\exp \lambda |f|^p] < \infty$ for some p and λ, then $E[\exp \alpha |f|^q] < \infty$ for any α and $q < p$.*

Proof:Let $S := \{x : |f(x)|^{p-q} \geq \alpha/\lambda\}$ and note that

$$\begin{aligned} E[\exp \alpha |f|^q] &= \left(\int_S + \int_{R^n \setminus S}\right) \exp \alpha |f(x)|^q\, \mu(dx) \\ &\leq \int_S \exp \lambda |f(x)|^p\, \mu(dx) + \int_{R^n \setminus S} \exp \alpha \left|\frac{\alpha}{\lambda}\right|^{\frac{q}{p-q}} \mu(dx) \\ &\leq \infty. \end{aligned}$$

Definition 3 *Given a function $f : R^n \to R^1$, a truncated version f_M is defined by*

$$f_M(x) = \begin{cases} f(x) & \text{if } |f(x)| \le M, \\ M & \text{if } f(x) > M, \\ -M & \text{if } f(x) < -M, \end{cases}$$

for a positive $M > 0$.

Note that $|f(x) - f_M(x)| \le |f(x)|$ for all $x \in R^n$.

IV. Approximation of Functions by MLPs

In this section, we state and prove the main thorem.

Theorem 4 *Let a be a continuous or nondecreasing sigmoidal function, and let K, μ, $\rho_{\lambda,p}$, and E be defined as in Definition 2. Let $f : R^n \to R^1$ be a Borel measurable function. If $\lambda > 0$, assume that $\rho_{\lambda c_p r,p}(f) < \infty$, where r is some number greater than one, and $c_p = 1$ for $0 < p \le 1$, and $= 2^{p-1}$ for $p > 1$. Given any $\varepsilon > 0$, there are an integer N, real constants c_i, b_i, $i = 1, \ldots, N$, and row n-vectors $w_i = [w_{i1}, \ldots, w_{in}]$, $i = 1, \ldots, N$, such that*

$$\rho_{\lambda,p}(f - s) < \varepsilon \tag{IV.1}$$

$$s(x) := \sum_{i=1}^{N} c_i a(w_i x + b_i). \tag{IV.2}$$

Proof:Case 1: $\lambda > 0$: By Theorem 1, there is a sequence

$$\{s_k(x) = \sum_{i=1}^{N} c_i(k)\, a(w_i(k)\, x + b_i(k))\} \tag{IV.3}$$

such that $\lim_{k\to\infty} s_k(x) = f_M(x)$ and $|s_k(x)| < M$ for almost every $x \in K$ and for any positive M. Notice that $|f_M(x) - s_k(x)| \le 2M$ and hence $E \exp[\lambda c_p r' |f_M - s_k|^p] \le \exp[2^p \lambda c_p r' M^p]$ for $r' = r/(r-1)$. By the dominated convergence theorem,

$$\lim_{k\to\infty} \rho_{\lambda c_p r',p}(f_M - s_k) = 0. \tag{IV.4}$$

Now notice that $|f(x) - f_M(x)|^p \le |f(x)|^p$ and hence $\rho_{\lambda c_p r,p}(f - f_M) \le \rho_{\lambda c_p r,p}(f)$. Since $\lim_{M\to\infty} |f(x) - f_M(x)|$ for all $x \in R^n$, by the dominated convergence thorem, the assumption that $\rho_{\lambda c_p r,p}(f) < \infty$ implies that

$$\lim_{M\to\infty} \rho_{\lambda c_p r,p}(f - f_M) = 0. \tag{IV.5}$$

We are now ready to show (IV.1) in the following: By an algebraic inequality and the Hölder inequality, we obtain for $\lambda > 0$,

$$\begin{aligned} \rho_{\lambda,p}(f - s_k) &= \rho_{\lambda,p}(f - f_M + f_M - s_k) \\ &\le \frac{1}{\lambda} \ln E[\exp(\lambda c_p |f - f_M|^p + \lambda c_p |f_M - s_k|^p)] \\ &\le \frac{1}{\lambda} \ln (E[\exp \lambda c_p r |f - f_M|^p])^{1/r} (E[\exp \lambda c_p r' |f_M - s_k|^p])^{1/r'} \\ &= \frac{1}{\lambda r} \ln E[\exp \lambda c_p r |f - f_M|^p] + \frac{1}{\lambda r'} \ln E[\exp \lambda c_p r' |f_M - s_k|^p] \end{aligned}$$

It follows from (IV.4) and (IV.5) that for any $\varepsilon > 0$, there are M and k such that $\rho_{\lambda,p}(f - s_k) < \varepsilon$. This completes the proof for $\lambda > 0$.

Case 2: $\lambda < 0$. By Theorem 1, there is a sequence

$$\{s_k(x) = \sum_{i=1}^{N} c_i(k)\, a(w_{i\cdot}(k)\, x + b_i(k))\} \tag{IV.6}$$

such that $\lim_{k\to\infty} s_k(x) = f(x)$ for almost all $x \in K$. Note that $0 < \exp\lambda|f - s_k|^p < 1$. By the dominated convergence theorem, $\lim_{k\to\infty} \rho_{\lambda,p}(f - s_k) = 0$, completing the proof for $\lambda < 0$.

V. Approximation of functions by RBFs

We notice that the proof of the main theorem in the preceding section is based on Theorem 1 concerning the uniform approximation by MLPs. Theorem 2 is analogous to Theorem 1 and thus provides a basis for extension of Theorem 4. Since the proof of the extension is almost verbatim the same, we only state the extended theorem for risk-sensitive approximation of functions by RBFs in the following.

Theorem 5 *Let $a(r, w_0)$ be defined as in Theorem 2, and let K, μ, $\rho_{\lambda,p}$, and E be defined as in Definition 2. Let $f : R^n \to R^1$ be a Borel measurable function. If $\lambda > 0$, assume that $\rho_{\lambda c_p r,p}(f) < \infty$, where r is some number greater than one, and $c_p = 1$ for $0 < p \leq 1$, and $= 2^{p-1}$ for $p > 1$. Given any $\varepsilon > 0$, there are an integer N, real constants w_0, c_i, w_{ij}, $i = 1, \ldots, N$, $j = 1, \ldots, n$, such that*

$$\rho_{\lambda,p}(f - s) \;<\; \varepsilon \tag{V.1}$$

$$s(x) \;:=\; \sum_{i=1}^{N} c_i a\left(\sqrt{\textstyle\sum_{j=1}^{n}(x_j - w_{ij})^2}, w_0\right). \tag{V.2}$$

VI. Conclusion

This paper shows that under mild conditions, functions can be approximated to an arbitrary accuracy with respect to the general risk-sensitive functionals by MLPs and RBFs. These results are mainly based on these neural networks' capability to uniformly approximate continuous functions arbitrarily closely on a compact support.

The results can therefore be extended to other neural network paradigms that have the same uniform approximation capability. Among such neural networks paradigms are variations of the MLPs and RFBFs considered here and the high-order neural networks [7, 6].

VII. REFERENCES

[1] G. Cybenko. Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals and Systems*, 2:303–314, 1989.

[2] K. Funahashi. On the approximate realization of continuous mappings by neural networks. *Neural Networks*, 2:183–192, 1989.

[3] K. Glover and J. C. Doyle. State-space formulae for all stabilizing controllers that satisfy an h-infinity norm bound and relations to risk-sensitivity. *Systems Control Letters*, 11:167–172, 1988.

[4] E. Hartman, J. Keeler, and J. Kowalski. Layered neural networks with gaussian hidden units as universal approximations. *Neural Computation*, 2:210–215, 1990.

[5] B. Hassibi, A. H. Sayed, and T. Kailath. $h^i nfty$ optimality of the LMS algorithm. *IEEE Transactions on Signal Processing*, 44:267–280, 1996.

[6] M. H. Hassoun. *Fundamentals of Artificial Neural Networks.* MIT Press, Cambridge, Massachusetts, 1995.

[7] S. Haykin. *Neural Networks.* Macmillan College Publishing Company, New York, New York, 1994.

[8] K. Hornik, M. Stinchcombe, and H. White. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2:359–366, 1989.

[9] D. H. Jacobson. Optimal stochastic linear systems with exponential performance criteria and their relation to deterministic games. *IEEE Transactions on Automatic Control*, AC-18-2:124–131, 1973.

[10] J. T. Lo. Finite dimensional sensor orbits and optimal nonlinear filtering. *IEEE Transactions on Information Theory*, IT-18-15:583–588, 1972.

[11] R. S. Mangoubi. Robust estimation and failure detection for linear systems. Technical Report Ph.D. Dissertation, Department of Aeronautics and Astronautics, MIT, Cambridge, MA, 1995.

[12] E. Parzen. On estimation of a probability density function and mode. *Annals of Mathematical Statistics*, 33:1065–1076, 1962.

[13] T. Poggio and F. Girosi. A theory of networks for approximation and learning. Technical Report Technical Report AI Memo No. 1140, Department of Mathematics, MIT, Cambridge, MA, 1989.

[14] J. Speyer, J. Deyst, and D. H. Jacobson. Optimaization of stochastic linear systems with additive measurement and process noise using exponential performance criteria. *IEEE Transactions on Automatic Control*, AC-19:358–366, 1974.

[15] J. L. Speyer, C.-H. Fan, and R. N. Banavar. Optimal stochastic estimation with exponential cost criteria. In *Proceedings of the 31st Conference on Decision and Control*, New York, New York, 1992. IEEE.

[16] P. Whittle. *Risk Sensitive Optimal Control.* Wiley, New York, New York, 1990.

Noisy Recurrent Neural Networks: Theory and Quantitative Performance Evaluation*

Oluseyi Olurotimi and Soumitra Das
Dept. of Electrical and Computer Engineering, MS 1G5
George Mason University
Fairfax, VA 22030
Phone: (703)993-1603
E-mail:seyi@bass.gmu.edu; soumitra@bass.gmu.edu

Abstract

The classical stochastic analog of the deterministic linear system in engineering is the linear system driven by white noise. This model is the basis of many important engineering methodologies in stochastic control, system identification, and signal estimation and classification. As the promise of artificial neural networks in modeling nonlinear systems continues to grow, the need for a stochastic analog with quantitative foundations for analysis and synthesis will increase. This papers reports our recent work in this direction, examining recurrent neural networks (RNN) driven by white noise.

We perform qualitative analysis establishing uniform boundedness of moments of the neuron states over time. To enable practical application in RNN design and evaluation, however, it is necessary to relate these properties to useful measures that can be estimated. We thus subsequently consider the RNN as a stochastic estimator of noisy dynamic patterns. We derive bias and variance measures for this estimator. This has significant practical implications, since neural network design is nonminimal in the sense that several different networks can be constructed to solve the same problem. The results in this paper allow the user to quantitatively evaluate given RNN's for noise performance. In addition, the designer can use these results to constrain the design space so that the achieved design satisfies performance specifications whenever possible.

We illustrate the application of these results to RNN design for temporal pattern generation.

Finally, we present the different, but analogous results for the discrete-time (so-called *time-lagged recurrent*) case.

1 Introduction and Motivation

1.1 Motivation

Suppose the reader were provided with the three recurrent neural network performance graphics of Figure 1. Each plot shows the performance of a different recurrent neural

*This work was supported in part by the National Science Foundation under Grant ECS-9209456.

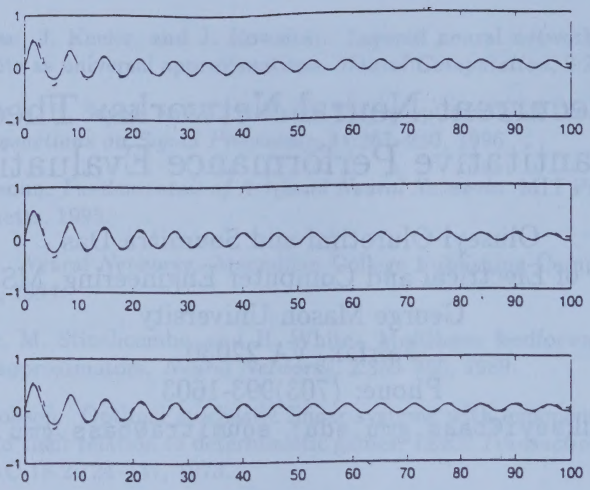


Figure 1: Comparisons of the output of three recurrent neural networks (solid line) with the target dynamic pattern (dashed line) when the recurrent neural networks are operating in a *noise-free* environment. Note that the RNN performances are quite comparable.

network (different weights, same neuron count) relative to a target dynamic pattern. The target pattern is indicated by the dashed line, while the recurrent neural network output is the solid line. All recurrent neural networks were operating in a noise-free environment. The reader is about equally likely to select any one of these three recurrent neural networks as an acceptable approximant for the target temporal pattern.

However, suppose we now examine Figure 2. In this situation where input noise is present in the recurrent neural network environment, it becomes clear that the topmost design is to be preferred, followed by the middle design, and the bottom design is the poorest.

The above illustration highlights an issue that is of major practical significance in the real-world use of artificial neural networks. One can readily identify situations where the above phenomenon can lead to significant problems in neural network design and evaluation:

1. *Neural networks for system identification and control.* When neural networks are inserted in place of the plant and/or controller, the difference in the responses to noisy input as illustrated in Figure 2 will cause variations in overall plant performance. In other words, the same application using identically performing (noise-free) recurrent neural networks will yield different evaluations when the control system operates in the presence of noisy inputs or disturbances. Note that while training with noise may alleviate some of these problems, the phenomenon of local minima is not eliminated. Different classes of degenerate RNN solutions that appear identical in the noise-free case will exhibit varying behavior in the presence of noise.

2. *Transferability of results/Repeatability of experiments.* A key aspect in validating scientific and engineering claims is the ability of other investigators to replicate the

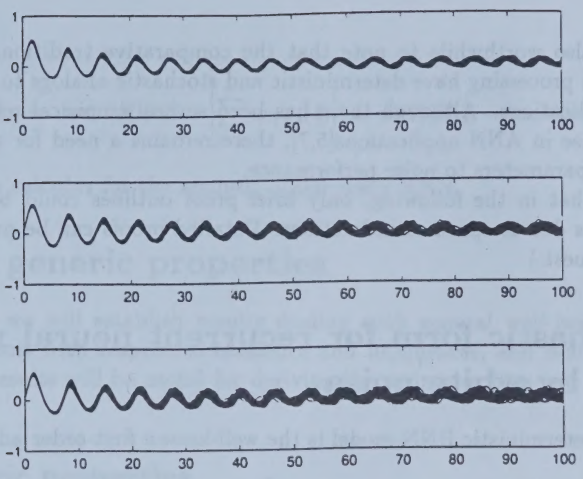


Figure 2: Comparisons of the output of three recurrent neural networks when operating in a *noisy* environment. Note that the recurrent neural network performances are now easily differentiated from each other.

corresponding experiments, and validate the claims made. The above illustration indicates that neural networks trained at different times or in different places, while appearing identical in performance in the absence of noise, may not yield consistent claims in the presence of noise.

Since a large proportion of real-world applications occur in the presence of noise, it becomes clear that an important practical issue is how to evaluate and select between available designs. A closely related issue is that of tailoring recurrent neural network design to yield improved noise robustness properties. These are the issues addressed by this paper.

1.2 Introduction

Recurrent neural networks (RNN), as well as multilayer feedforward neural networks (MNN) are both forms of artificial neural networks (ANN). Currently, MNN are more well-understood than RNN.

Much empirical success has been reported using MNN for nonlinear mapping applications in areas such as signal processing, pattern recognition, and control. It is now known that such networks can approximate not only any continuous maps but also derivatives of such maps[1,2].

RNN are a form of ANN in which, unlike MNN, feedback connections are allowed. Consequently, in contrast to MNN, RNN exhibit time-varying, dynamic behavior. It has been shown that a continuous time RNN can approximate a given dynamical system arbitrarily closely[3]. Consequently, RNN are increasingly being proposed for use in many areas such as general control and signal processing applications[4–6].

In engineering and other applications noisy input to the system is common. Therefore it is prudent to anticipate the presence of *uncontrolled* noise in applications where RNN's

are used. It is also worthwhile to note that the comparative traditional methods in control and signal processing have deterministic and stochastic analogs to accommodate this range of applications. Although there has been several empirical references about the impact of noise in ANN applications[5,7], there remains a need for specific results relating network parameters to noise performance.

(We remark that in the following, only brief proof outlines could be supplied for the various results due to space considerations. Detailed proofs can be provided by the authors upon request.)

2 A stochastic form for recurrent neural networks driven by white noise

Our underlying deterministic RNN model is the well-known first-order additive form:

$$\dot{x}_i(t) = -a_i x_i(t) + g_i(\sum_{j=1}^{n} w_{ij} x_j(t)) + I_i(t) \tag{1}$$

where $x_i(t)$ is the output of the i^{th} neuron, a_i is a time constant or a decay constant, w_{ij} is a weight connecting the j^{th} neuron to the i^{th} neuron, w_{ij} is not required to be symmetric, $I_i(t)$ is a possibly time-varying input to i^{th} neuron, and $g_i(\cdot)$ is a bounded, monotone nondecreasing function.

We consider an RNN differential equation driven by white noise:

$$\dot{x}_i(t) = -a_i x_i(t) + g_i(\sum_{j=1}^{n} w_{ij} x_j(t)) + I_i(t) + \sqrt{T_i}\eta_i(t),$$

where T_i is a nonnegative constant diffusion factor, and we use $\eta_i(t)$ as a notation for white noise.

The general form for passage of a vector differential equation driven by white noise to a vector stochastic differential equation is given bin[8]. We find in our case,

$$dx_i(t) = (-a_i x_i(t) + g_i(\sum_{j=1}^{n} w_{ij} x_j(t)) + I_i(t))dt + \sqrt{T_i} dW_i(t) \tag{2}$$

where $i = 1, 2...n$, and W_i is the Wiener process or Brownian motion perturbing the i^{th} neuron, and T_i is a nonnegative diffusion factor. Note that unlike applications that focus on either improving learning, or on stochastic optimization, we do not assume control of the diffusion factor T_i.

Certain assumptions are often implicit in neural network use and design. We formalize several of them below for ease of reference in the subsequent development.

A1. a_i is a nonnegative, bounded constant.

A2. $g_i(\cdot)$ is bounded, twice differentiable, and monotone nondecreasing, with $g_i(0) = 0$.

A3. w_{ij} is bounded and constant.

A4. $I_i(t)$ is a uniformly bounded function of time.

A5. T_i is a bounded nonnegative constant.

A6.

$$|x_i(t)| \to \infty \Longrightarrow |\xi_i(t)| = \left|\sum_{j=1}^{n} w_{ij} x_j(t)\right| \to \infty;\ |\xi_i(t)| \to \infty \Longleftrightarrow |\tilde{\xi}_i(t)| = \left|\sum_{j=1}^{n} \frac{a_j}{a_i} w_{ij} x_j(t)\right|.$$

We use Itô's calculus for the analysis of our noisy RNN.

3 Some generic properties

In this section we will establish results dealing with general well-behavedness of the stochastic systems with respect to existence and uniqueness, and boundedness of moments. These results will be useful for deriving the estimator bias and variance bounds of Section 4.

3.1 Generic properties

3.1.1 A useful fact

We state the following well-known fact (*cf.* [9,10]) for easy reference

Lemma 3.1 *Consider the dynamical system*

$$\dot{q}_i(t) = -aq_i(t) + M(\boldsymbol{q}(t)),\ i = 1, 2, \ldots, n,$$

where $q_i(t) \in \mathcal{R}$, $a > 0, a \in \mathcal{R}$, $\boldsymbol{q} \in \mathcal{R}^n$, $-\infty < M_{min} \leq M(\boldsymbol{q}(t)) \leq M_{max} < \infty$. *Then we have*

$$\frac{M_{min}}{a} \leq q_i(t) \leq \frac{M_{max}}{a}.$$

Proof: See analogous proofs in [9,10].■

3.1.2 Existence and uniqueness of solutions

The standard existence and uniqueness statements for stochastic differential equations can be found in appropriate references[8]. They are not restated here for space considerations. They can be readily verified for our stochastic recurrent neural network model.

3.1.3 Boundedness of Moments

Although the earlier existence and uniqueness conditions ensure *pointwise* boundedness of the first and second moments, we are interested in finding out whether these moments, although pointwise bounded, grow unbounded with time as the dynamic system evolves. We show that the first and second moments of the stochastic RNN are uniformly bounded over time. To enable this, we derive the associated dynamical systems describing the evolution of the moments of the trajectory.

Proposition 3.1 (Uniform boundedness of the first moment) *Consider the stochastic recurrent neural network*

$$dx_i(t) = (-a_i x_i(t) + g_i(\sum_{j=1}^{n} w_{ij} x_j(t)) + I_i(t))dt + \sqrt{T_i} dW_i(t)$$

subject to assumptions A1–A6 of Section 2.

The first moment, $E(x_i(t))$, *of the trajectory* $x_i(t)$ *is uniformly bounded over time.*

Proof:

The outline of the proof is as follows:

Integrate the stochastic recurrent neural network and take expectations. The noise term vanishes. After invoking Fubini's theorem and taking derivatives, we obtain an ordinary differential equation in the expected values similar to the deterministic recurrent neural network equation. Then Lemma 3.1 yields the final result. ■

The following is a key lemma which is necessary for establishing subsequent results.

Lemma 3.2 *Consider the stochastic recurrent neural network*

$$dx_i(t) = (-a_i x_i(t) + g_i(\sum_{j=1}^{n} w_{ij} x_j(t)) + I_i(t))dt + \sqrt{T_i} dW_i(t)$$

subject to assumptions A1–A6 of Section 2.

$E(x_i(t) g_i(\xi_i(t)))$ is uniformly bounded over time, where $\xi_i = \sum_{j=1}^{n} w_{ij} x_j(t)$.

Proof:

The outline of the proof is as follows:

We use Itô's differentiation formula for $x_i(t) g_i(\xi_i(t))$. Taking expectations, using Fubini's theorem, and the vanishing property of certain stochastic integrals in the Itô (forward differencing) calculus, as well as Assumptions A1–A6, we obtain a form from which Lemma 3.1 yields the final result. ■

The following establishes the uniform boundedness of the second moment.

Proposition 3.2 (Uniform boundedness of the second moment) *Consider the stochastic recurrent neural network*

$$dx_i(t) = (-a_i x_i(t) + g_i(\sum_{j=1}^{n} w_{ij} x_j(t)) + I_i(t))dt + \sqrt{T_i} dW_i(t)$$

subject to Assumptions A1–A6 of Section 2.

The second moment, $E(x_i^2(t))$, of the trajectory $x_i(t)$ is uniformly bounded over time.

Proof:

The outline of the proof is as in the proof of Lemma 3.2, along with the fact that we also need to use the same lemma before Lemma 3.1 can be invoked for the final result. ■

4 Stochastic RNN as an estimator of time-varying patterns

The earlier results have established the fact that the stochastic variations of the trajectory traced by the noisy RNN will be uniformly bounded for all time. Although these boundedness results are an important step in the analysis of noisy RNN, they may not always be amenable to practical RNN design. This is because, apart from the decay constant a_i, the effect of critical recurrent neural network parameters (namely the weights) are not revealed.

In this section, by considering the stochastic RNN as an estimator of time-varying patterns, we are able to derive more practical design and evaluation measures for noisy

RNN. In particular, we are able to relate practically relevant quantities such as the "bias*" and variance of this estimator to actual RNN parameters.

In the following, we consider that use of the RNN used in a noisy environment implicitly defines it as an estimator of the pattern being generated. We omit a formal definition statement due to space considerations.

4.1 RNN stochastic estimator bias and variance

Proposition 4.3 (RNN estimator bias) *Consider the stochastic RNN estimator. Let* $\boldsymbol{\mathcal{W}}$ *be the* $n \times n$ *weight matrix with* (i,j) *element* w_{ij}, $\boldsymbol{A}$ *the* $n \times n$ *diagonal matrix with* i^{th} *entry* a_i, *and* $\boldsymbol{\Delta}(E\boldsymbol{\xi}(t))$ *the* $n \times n$ *diagonal matrix with* i^{th} *entry* $\alpha_i(E\xi_i(t))$, *where*

$$0 < \alpha_i(E\xi_i(t)) = \frac{g_i\left(\sum_{j=1}^n w_{ij} E x_j(t)\right) - g_i\left(\sum_{j=1}^n w_{ij}\chi_j(t)\right)}{\sum_{j=1}^n w_{ij} b_j(t)} \leq g_{i_{max'}},$$

$$\xi_i(t) = \sum_{j=1}^n w_{ij} x_j(t).$$

Let $\boldsymbol{b}(t)$ *be the vector of biases at time* t. *Let* $\lambda_{min}[\boldsymbol{M}]$ *and* $\lambda_{max}[\boldsymbol{M}]$ *denote the smallest and largest pointwise eigenvalue of* $\boldsymbol{M}$ *respectively (we shall also alternately denote these by* $\lambda_{\boldsymbol{M}_{min}}$ *and* $\lambda_{\boldsymbol{M}_{max}}$ *respectively when more convenient), then*

$$\|\boldsymbol{P}\boldsymbol{b}^{zi}(t)\| \leq \|\boldsymbol{P}\boldsymbol{b}_0\| e^{\frac{1}{2}\int_0^t \mu_{\boldsymbol{P}}(\tau)\lambda_{\boldsymbol{M}_{max}}(\tau)d\tau}, \quad t \geq 0$$

$$\boldsymbol{M}(t) = \boldsymbol{P}^T\boldsymbol{P}\left(-\boldsymbol{A} + \boldsymbol{\Delta}(E\boldsymbol{\xi}(t))\boldsymbol{\mathcal{W}}\right) + \left(-\boldsymbol{A} + \boldsymbol{\Delta}(E\boldsymbol{\xi}(t))\boldsymbol{\mathcal{W}}\right)^T\boldsymbol{P}^T\boldsymbol{P},$$

where $\boldsymbol{b}^{zi}(t)$ *is the part of the bias* $\boldsymbol{b}(t)$ *associated with the initial bias* $\boldsymbol{b}(0) = \boldsymbol{b}_0$[†], $\boldsymbol{P}$ *is a user-specified output matrix, and*

$$0 < \mu_{\boldsymbol{P}}(t) = \frac{\|\boldsymbol{b}^{zi}(t)\|^2}{\|\boldsymbol{P}\boldsymbol{b}^{zi}(t)\|^2} < \infty,$$[‡]

Furthermore if

$$\lambda_{max}\left[-2\boldsymbol{A} + \boldsymbol{\Delta}(E\boldsymbol{\xi}(t))\boldsymbol{\mathcal{W}} + \boldsymbol{\mathcal{W}}^T\boldsymbol{\Delta}(E\boldsymbol{\xi}(t))\right] < 0, \; t \geq 0$$

then an upper bound on the bias is given by

$$\lim_{t\to\infty} \|\boldsymbol{b}(t)\| \leq \frac{\nu\sqrt{n}(g_{max} - g_{min})}{\lambda},$$

where $0 < \nu < \infty$, $0 < \lambda < \infty$ *are associated with the rate of decay of a state transition matrix associated with the estimator.*

*Note that "bias" in a neural network context often applies to a weight with a constant input. However, "bias" in the estimator sense, and as used here, is the difference between the statistical expectation of the (RNN) estimator output and the true, target output.

†The superscript *zi* indicates that this is the unforced, homogeneous, or "zero-input" solution.

‡We make the necessary practical assumption that the output matrix has been chosen such that this ratio remains finite.

Proof:

The outline of the proof is as follows:

The recurrent neural network nonlinearity is expanded in a Taylor series about the expected values of the neuron states. Then noting that this nonlinearity is bounded and monotone nondecreasing, the mean value theorem leads to a specialized equivalent linear time-varying dynamic system for the expected values. Lyapunov and other stability techniques are successfully employed to achieve uniform exponential stability results[11]. The result implies that the RNN with the smallest $\lambda_{M_{max}}(t)$ is best in noise. ■

Proposition 4.4 (RNN estimator variance) *Consider the stochastic RNN estimator. If*

$$-a_i + w_{ii}\alpha_{i\sigma}(\xi_i(t)) < 0,$$

then

$$\frac{\Psi_{imin}}{2\left(a_i - w_{ii}\alpha_{i\sigma}(\xi_i(t))\right)} \leq \sigma_i^2(t) \leq \frac{\Psi_{imax}}{2\left(a_i - w_{ii}\alpha_{i\sigma}(\xi_i(t))\right)}.$$

where $\sigma_i^2(t)$ is the variance of the estimator, and

$$0 < \alpha_{i\sigma}(\xi_i(t)) = \frac{E\left(\frac{g_i(\xi_i(t))}{\xi_i(t)} x_i^2(t)\right)}{E\left(x_i^2(t)\right)} < g_{imax'}.$$

Proof:

The outline of the proof is similar to that of Proposition 4.3, except that instead of the stability methods, Lemmas 3.2 and 3.1 are used to obtain the stated bounds. The result implies that the RNN with the smallest w_{ii} is best in noise. ■

5 Simulations

In this section we describe the training of three different RNN's temporal pattern generation, and confirm that the off-line theoretical predictions concerning relative noise performance are corroborated.

We desire to train an RNN as an "estimator" of a dynamical system generating Bessel functions $J_\nu(t)$. That is, for some neuron i in the RNN, we have

$$x_i \simeq J_\nu(t),$$

and where the approximation remains reasonable during subsequent pattern recall in the presence of noise.

5.1 Recurrent neural network designs

Pearlmutter's algorithm was used for training the RNN's [12]. (Some straightforward scaling modifications to the preceding formulas are necessary to use the model in this algorithm). A network size of six neurons was arbitrarily chosen. This consisted of one output unit. The other five neurons may be viewed as "hidden" units. The index number of the output neuron is six, *i.e.* the output variable is $x_6(t)$. We used a hyperbolic tangent for the sigmoidal function, i.e., $g_i(z) = \tanh(z)$. The target output for training was generated by sampling a Bessel function of the first kind of order 1 at 1001 evenly spaced points in the arbitrarily chosen interval $0 \leq t \leq 100$. To train the time constant

RNN	$\overline{\omega_i(t)}$	$\overline{\lambda_{max}[\boldsymbol{M}(t)]}$
#1	-7.6357	.4300
#2	-1.8035	.6599
#3	0.2663	.8748

Table 1: Recurrent neural network designs and corresponding time-averaged measures of performance.

a_i, gradient descent in $log(a_i)$ was used as suggested by Pearlmutter [13]. The error goal was set at .01.

The corresponding time-averaged measures of performance $\overline{\lambda_{max}[\boldsymbol{M}(t)]}$ and $\overline{\omega_i(t)}$ (specified in the earlier bias and variance results) are shown in Table 1.

¿From the table above, we notice that both $\overline{\omega_i(t)}$ and $\overline{\lambda_{max}[\boldsymbol{M}(t)]}$ are lowest for RNN#1, followed by RNN#2, and RNN#3, respectively. The predicition is therefore that RNN#1 will be the preferred design since it should exhibit the greatest degree of noise robustness, followed by RNN#2 and RNN#3, respectively.

This prediction is checked with actual simulation in the presence of noise in the next section.

5.2 Recurrent neural network performance

RNN#1, RNN#2, and RNN#3 were simulated in the absence of noise, each generating a Bessel function. Then the same simulations were performed with white noise having diffusion factor 0.0001. Figures 3 and 4 depict zero-noise and noisy outputs of the RNN's alongside their respective estimator biases and variances, respectively.

The first column of Figure 3 shows the deterministic (zero-noise) outputs of the RNN (solid lines) along with the desired outputs (dashed lines). Figure 3*(a)* corresponds to RNN#1, Figure 3*(b)* corresponds to RNN#2, while Figure 3*(c)* corresponds to RNN#3. We observe that the RNN performances are quite comparable. However, when white noise is introduced, the performances as shown in Figure 3*(d)*,*(e)*,*(f)* respectively clearly indicate the superiority of RNN#1, followed by RNN#2, and then RNN#3. This agrees with the earlier predictions. The time patterns of the bias, shown in Figure 3*(g)*,*(h)*,*(i)* respectively also follow the same trend.

The first two columns of Figure 4 repeat the information of Figure 3 for ease of visual comparison. The third column displays the respective time patterns of the variance, maintaining the predicted trend.

6 The Discrete-Time Case

Analogous results have also been obtained for the discrete-time (or so-called *time-lagged recurrent*) case. Due to space considerations, we state only Propositions regarding the bias and variance measures.

Proposition 6.5 (Discrete-time RNN estimator bias) *Consider the stochastic discrete-time RNN estimator. Let* $\mathcal{W}$ *be the* $n \times n$ *weight matrix with* (i,j) *element* w_{ij}*, and*

$\boldsymbol{\Delta}(E\boldsymbol{\xi}(k))$ *the* $n \times n$ *diagonal matrix sequence with* i^{th} *entry* $\alpha_i(E\xi_i(k))$, *where*

$$\begin{aligned} 0 < \alpha_i(E\xi_i(k)) &= \frac{g_i\left(\sum_{j=1}^n w_{ij} E x_k(t)\right) - g_i\left(\sum_{j=1}^n w_{ij}\chi_j(k)\right)}{\sum_{j=1}^n w_{ij} b_j(k)} \leq g_{imax'}, \\ \xi_i(k) &= \sum_{j=1}^n w_{ij} x_j(k). \end{aligned}$$

Let $\boldsymbol{b}(k)$ *be the vector of the bias sequence. Let* $\lambda_{min}[\boldsymbol{M}]$ *and* $\lambda_{max}[\boldsymbol{M}]$ *denote the smallest and largest pointwise eigenvalue of* $\boldsymbol{M}$ *respectively (we shall also alternately denote these by* $\lambda_{\boldsymbol{M}_{min}}$ *and* $\lambda_{\boldsymbol{M}_{max}}$ *respectively when more convenient), then*

$$\begin{aligned} \|\boldsymbol{P}\boldsymbol{b}^{zi}(k)\| &\leq \|\boldsymbol{P}\boldsymbol{b}^{zi}(0)\| \prod_{j=0}^{k-1} \mu_{\boldsymbol{P}}^{\frac{1}{2}}(j) \lambda_{\boldsymbol{M}_{max}}^{\frac{1}{2}}(j), \ k = 0, 1, 2, \ldots \\ \boldsymbol{M}(k) &= \boldsymbol{\mathcal{W}}^T \boldsymbol{\Delta}(E\boldsymbol{\xi}(k)) \boldsymbol{P}^T \boldsymbol{P} \boldsymbol{\Delta}(E\boldsymbol{\xi}(k)) \boldsymbol{\mathcal{W}}. \end{aligned}$$

where $\boldsymbol{b}^{zi}(k)$ *is the part of the bias sequence* $\boldsymbol{b}(k)$ *associated with the initial bias* $\boldsymbol{b}(0) = \boldsymbol{b}_0$[§], $\boldsymbol{P}$ *is a user-specified output matrix, and*

$$0 < \mu_{\boldsymbol{P}}(k) = \frac{\|\boldsymbol{b}^{zi}(k)\|^2}{\|\boldsymbol{P}\boldsymbol{b}^{zi}(k)\|^2} < \infty,$$[¶]

Furthermore if

$$0 \leq \lambda_{max}\left[\boldsymbol{\mathcal{W}}^T \boldsymbol{\Delta}(E\boldsymbol{\xi}(k)) \boldsymbol{\Delta}(E\boldsymbol{\xi}(k)) \boldsymbol{\mathcal{W}}\right] < 1, \ k = 0, 1, 2, 3, \ldots.$$

then an upper bound on the bias is given by

$$\lim_{k \to \infty} \|\boldsymbol{b}(k)\| \leq \frac{\nu\sqrt{n}(g_{max} - g_{min})}{1 - \lambda},$$

where $0 < \nu < \infty$, $0 \leq \lambda < 1$ *are associated with the rate of decay of a state transition matrix associated with the estimator.*

Proposition 6.6 (RNN estimator variance) *Consider the stochastic discrete-time RNN estimator. If*

$$w_{ii}^2 \alpha_{i\sigma}^2\left(\xi_i(k)\right) < 1,$$

then

$$0 \leq \sigma_{i,\infty}^2 \leq \frac{\Psi_{imax}}{1 - w_{ii}^2 g_{i\,max'}^2}.$$

where $\sigma_{i,\infty}^2$ *is the asymptotic value of the variance sequence of the estimator, and*

$$0 < \alpha_{i\sigma}(\xi_i(k)) = \left(\frac{E\left(\frac{g_i^2(\xi_i(k))}{\xi_i^2(k)} x_i^2(k)\right)}{E\left(x_i^2(k)\right)}\right)^{\frac{1}{2}} < g_{imax'}.$$

[§]The superscript zi indicates that this is the unforced, homogeneous, or "zero-input" solution.

[¶]We make the necessary practical assumption that the output matrix has been chosen such that this ratio remains finite.

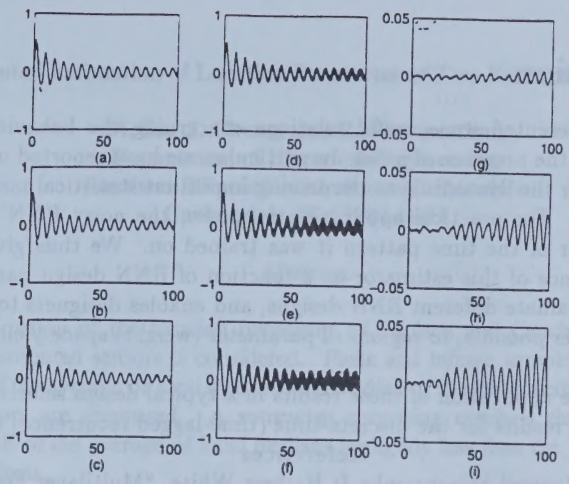


Figure 3: *(a).* RNN #1 zero-noise output (solid line); desired output (dashed line). *(b).* RNN #2 zero-noise output (solid line); desired output (dashed line). *(c).* RNN #3 zero-noise output (solid line); desired output (dashed line). *(d).* RNN #1 noisy output. *(e).* RNN #2 noisy output. *(f).* RNN #3 noisy output. *(g).* RNN #1 bias as a function of time. *(h).* RNN #2 bias as a function of time. *(i).* RNN #3 bias as a function of time.

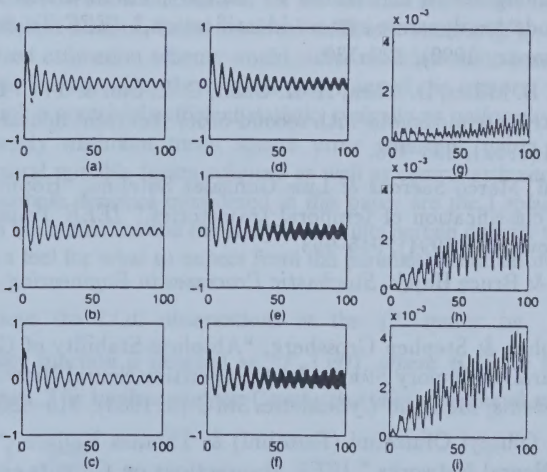


Figure 4: *(a).* RNN #1 zero-noise output (solid line); desired output (dashed line). *(b).* RNN #2 zero-noise output (solid line); desired output (dashed line). *(c).* RNN #3 zero-noise output (solid line); desired output (dashed line). *(d).* RNN #1 noisy output. *(e).* RNN #2 noisy output. *(f).* RNN #3 noisy output. *(g).* RNN #1 variance as a function of time. *(h).* RNN #2 variance as a function of time. *(i).* RNN #3 variance as a function of time.

7 Conclusion

This paper has presented some useful relations concerning the behavior of recurrent neural networks in the presence of noise. In particular, we have reported uniform boundedness over time for the dynamic systems driving important statistical measures of noisy RNN performance. For practical application purposes, the noisy RNN is viewed as a nonlinear estimator of the time pattern it was trained on. We thus give estimates of the bias and variance of this estimator as a function of RNN design parameters. This enables users to evaluate different RNN designs, and enables designers to limit parameter search, whenever possible, to regions of parameter (weight) space yielding acceptable noise performance.

We outlined the application of these results in a typical design selection application, and also presented results for the discrete-time (time-lagged recurrence) case.

References

[1] Kurt Hornik, Maxwell Stinchcombe & Halbert White, "Multilayer Feedforward Networks are Universal Approximators," *Neural Networks* 2 (1989), 359–366.

[2] Kurt Hornik, Maxwell Stinchcombe & Halbert White, "Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks," *Neural Networks* 3 (1990), 551–560.

[3] Ken-Ichi Funahashi & Yuichi Nakamura, "Approximation of Dynamical Systems by Continuous Time Recurrent Neural Networks," *Neural Networks* 6 (1993), 801–806.

[4] "NIPS 95 Workshop: Neural networks for signal processing," in *Advances in Neural Information Processing Systems 7*, December, 1995.

[5] Si-Zhao Qin, Hong-Te Su & Thomas J. McAvoy, "Comparison of four neural net learning methods for dynamic system identification," *IEEE Transactions on Neural Networks* 3 (January 1992), 122–130.

[6] C. L. Giles, C. B. Miller, D. Chen, H. H. Chen, G. Z. Sun & Y. C. Lee, "Learning and extracting finite state automata with second-order recurrent neural networks," *Neural Computation* 4 (1992), 393–405.

[7] Hugues Bersini, Marco Saerens & Luis Gonzalez Sotelino, "Hopfield net generation, encoding and classification of temporal trajectories," *IEEE Transactions on Neural Networks* 5 (November 1994), 945–953.

[8] Eugene Wong & Bruce Hajek, *Stochastic Processes in Engineering Systems*, Springer-Verlag, 1985.

[9] Michael A. Cohen & Stephen Grossberg, "Absolute Stability of Global Pattern Formation and Parallel Memory Storage by Competitive Neural Networks," *IEEE Transactions on Systems, Man and Cybernetics* SMC-13 (1983), 815–826.

[10] Amir Dembo, Oluseyi Olurotimi (Farotimi) & Thomas Kailath, "High-Order Absolutely Stable Neural Networks," *IEEE Transactions on Circuits and Systems* 38 (January 1991).

[11] Wilson J. Rugh, *Linear System Theory*, Prentice Hall, Englewood Cliffs, NJ, 1996.

[12] Barak A. Pearlmutter, "Learning state space trajectories in recurrent neural networks," *Neural Computation* 1 (1989), 263–269.

[13] Barak A. Pearlmutter, "Temporally continuous vs. Clocked Networks," in *Neural Networks in Robotics*, Kenneth Goldberg & George A. Bekey, eds., Kluwer Academic, 1992, 237–252.

Distributed Estimation of Location Parameter of Two Example Densities*

M. Su, J. Cheng, and R. Viswanathan
Department of Electrical Engineering
Southern Illinois University at Carbondale
Carbondale, IL 62901-6603

Abstract

Distributed estimation of the location parameter of Laplace and Cauchy densities using a collection of distributed sensors is considered. Finite and infinite sample mean square error performances of the average of local medians, the median of the local medians, and the central median estimators are compared. A somewhat surprising result is that, for the Laplace density, the MSE of the average of local medians is slightly less than the MSE of the median of the local medians.

1. Introduction

We consider the estimation of a nonrandom location parameter of a density function using grouped data. Such problems arise when estimating a parameter using data from a collection of sensors that may be geographically dispersed[1-4]. Assuming that the i^{th} sensor of an n-sensor collection forms an estimate τ_i based on its data, we investigate the mean square error (MSE) performances of several schemes that obtain final (global) estimates by combining (fusing) these local estimates, $(\tau_1,..,\tau_n)$. Each sensor receives a set of *i.i.d* samples and all the sensors' samples are statistically independent of each other. If a density function admits a nontrivial sufficient statistic for the location parameter, then it is possible to obtain a global sufficient statistic based on local sufficient statistics[5]. In such cases, an optimum decentralized estimation scheme would suffer no information loss as compared to a central estimator that has access to the original data from all the sensors. However, when a density does not admit a nontrivial sufficient statistic, typically no uniformly minimum variance estimator or uniformly minimum mean square error estimator exists. In such cases, performances of several possible fusion schemes as well as central estimators will have to be compared. Two example densities considered in this paper are the Laplace and the Cauchy densities. Although the numerical and the analytical results pertain directly to these two cases, this study provides a feel for what to expect from the parameter estimation procedures based on grouped data.

Let us denote the *i.i.d.* observations at the i^{th} sensor by $X_1^{(i)},..X_{k_i}^{(i)}$. The corresponding density function is denoted as $f_X(x|\theta)$, where θ is the location parameter that is to be estimated. The Laplace and the Cauchy densities considered are of the following form:

$$f_L(x|\theta) = \frac{1}{2\lambda} e^{-\frac{|x-\theta|}{\lambda}} \qquad (1)$$

* This work was supported by BMDO and managed by the Office of Naval Research under contract N00014-94-1-0736.

$$f_C(x|\theta) = \frac{1}{\pi\lambda\left(1+\left(\frac{x-\theta}{\lambda}\right)^2\right)} \qquad (2)$$

Here it is assumed that the scale parameter λ is known and is taken to be 1 without any loss of generality. The estimation of θ of Cauchy density, when λ is unknown, is briefly considered in [6]. Throughout the discussion, except in section 2.3, it is assumed that each sensor receives the same number, $k_i = k$, of random samples. For these situations, it is convenient to denote the data at the i^{th} sensor as $(X_{(i-1)k+1}, X_{(i-1)k+2}.., X_{ik})$, i=1,2,..,n and the corresponding local estimate of θ as τ_i. A global estimate is denoted as τ_0. Two decentralized estimators considered are, Scheme 1, a linear combination of local estimates:

$$\tau_0 = \sum_{i=1}^{n} c_i \tau_i \qquad (3)$$

and Scheme 2, an order statistic of local estimates:

$$\tau_0 = \text{OS}\,(\tau_1,..,\tau_n) \qquad (4)$$

The performances of (3) and (4) are compared with the central median estimator given by

$$\tau_0 = \text{median}\,(X_1, X_2,.., X_{nk}) \qquad (5)$$

where median denotes the middle order statistic of the observations. A local estimator is assumed to be

$$\tau_i = \text{OS}\ (X_{(i-1)k+1}, X_{(i-1)k+2}.., X_{ik}) \qquad (6)$$

The choice of an order statistic for local estimation is motivated by the results that (1) the sample median is a reasonable estimator of location for both the densities and (2) the sample median has a smaller MSE than the arithmetic average, for the case of Laplace density. The best linear unbiased estimator based on order statistics (BLUE) has a somewhat smaller MSE than the estimator (6), but (6) has been considered because it is reasonably good and simple. For the same reason, we did not consider a BLUE based on $(\tau_1,..,\tau_n)$ for a global estimate, but only considered (3) and (4).

In section 2, we provide relevant equations for the evaluation of MSE of different schemes. In the same section, the effect of unequal distribution of samples to the sensors is also discussed. Relative asymptotic performances of various schemes, as sample size becomes unbounded, are considered in section 3. We present in section 4 a discussion of the numerical results obtained from the previous sections.

2. MSE of Different Estimators

With the location transformation

$$Y_j = X_j - \theta \qquad (7)$$

the density of Y_j is free of the parameter θ. Denote the density and the corresponding CDF of Y_j as $f(.)$ and $F(.)$, respectively. $f(y)$ is obtained from (1) as $f_L(y|\theta=0,\lambda=1)$ for Laplace density, and from (2) as $f_C(y|\theta=0,\lambda=1)$ for Cauchy density. By denoting the r^{th} order statistic of a random sample of size k from $(X_1,..,X_k)$ as $X_{(r:k)}$, and by using a similar notation for the order statistic of the $(Y_j, j=1,2,..,k)$ samples, we have the following relations,

$$E(X_{(r:k)}) = E(Y_{(r:k)}) + \theta \quad = u^{(1)}_{(r:k)} + \theta \qquad (8)$$

$$\mathrm{Var}\left(X_{(r:k)}\right) = \mathrm{Var}\left(Y_{(r:k)}\right) = u^{(2)}_{(r:k)} - \left[u^{(1)}_{(r:k)}\right]^2 \tag{9}$$

where

$$u^{(j)}_{(r:k)} = E\left(\left(Y_{(r:k)}\right)^j\right) \tag{10}$$

In addition, for a symmetric $f(.)$ [7, pp. 27],

$$u^{(1)}_{(r:k)} = -u^{(1)}_{(k-r+1:k)} \tag{11}$$

$$u^{(1)}_{((k+1)/2:k)} = 0 \qquad \text{if } k \text{ is odd} \tag{12}$$

Also, when each sensor has the same sample size k,

$$E(\tau_1) = E(\tau_2) = .. = E(\tau_n) \tag{13}$$

2.1 MSE of Scheme 1

From (3),(8), and (13), we have for the bias of the estimator (3),

$$\text{bias} = u^{(1)}_{(r:k)} \sum_{i=1}^{n} c_i + \theta\left(\sum_{i=1}^{n} c_i - 1\right) \tag{14}$$

Since the MSE of an estimator equals the sum of variance and squared bias, and since it is desirable that the MSE is independent of θ, we set

$$\sum_{i=1}^{n} c_i = 1 \tag{15}$$

and $r = \dfrac{k+1}{2}$, k being odd.

The MSE of Scheme 1 is then given by

$$MSE_1(\tau_0) = u^{(2)}_{(r:k)} \sum_{i=1}^{n} c_i^2 \tag{16}$$

Therefore, the weights $\{c_i\}$ can be chosen so as to minimize (16) with the constraint (15) satisfied. This yields

$$c_i = \frac{1}{n} \quad i = 1,2,..,n \tag{17}$$

The corresponding minimum MSE for Scheme 1 is given by

$$MSE_1(\tau_0) = \frac{u^{(2)}_{(r:k)}}{n} \tag{18}$$

For Laplace density, the above MSE is easily evaluated using the expression in [8]. For Cauchy density, numerical integration was employed to compute (18) (also see [9], for tabulated values of $u^{(2)}_{(r:k)}$, for various values of r and k).

2.2 MSE of Scheme 2

Denote the density and the CDF of $Y_{(r:k)}$ as $f_{r:k}(.)$ and $F_{r:k}(.)$, respectively. Then the density of an l^{th} order statistic of $(\tau_1,..,\tau_n)$, denoted as $g_{l:n}(.)$, is given by

$$g_{l:n}(\tau) = \frac{n!}{(l-1)!\,(n-l)!}\left(F_{r:k}(\tau)\right)^{l-1}\left(1-F_{r:k}(\tau)\right)^{n-l} f_{r:k}(\tau) \tag{19}$$

Each local estimator τ_i can be made unbiased by the choice of $r = \dfrac{k+1}{2}$, and an odd number k.

Since the ordering preserves location, the MSE of Scheme 2 can be expressed as

$$MSE_2(\tau_0) = E(\tau_0 - \theta)^2 = E\left(\tau_{(l:n)}\right)^2 \tag{20}$$

where $\tau_{l:n}$ has the density given by (19). Using (19), (20), and a numerical integration routine, the MSE has been evaluated.

2.3 MSE of Scheme 1 With Unequal Sensor Samples

Assuming that the i^{th} sensor has k_i samples, we consider the MSE performance of Scheme 1. For convenience, we denote the corresponding MSE as $MSE_3(\tau_0)$. Proceeding as in the equal sample case, and with the constraint (15), we obtain

$$MSE_3(\tau_0) = \sum_{i=1}^{n} c_i^2 v_i + \left[\sum_{i=1}^{n} c_i u^{(1)}_{(r_i:k_i)}\right]^2 \tag{21}$$

where

$$v_i = \mathrm{Var}\left(Y_{(r_i:k_i)}\right) \tag{22}$$

The coefficients $\{c_i\}$ can be chosen so as to minimize (21) subject to the constraint (15). By a straight forward procedure it is possible to arrive at the solution, which is given in terms of simple matrix multiplications[6]. In fact, the structure of the equations leading to the optimal solution admits an analytical expression for a required matrix inversion, thereby simplifying numerical computations[6].

3. Asymptotic MSE

In this section we evaluate the asymptotic MSE of different schemes as sample size becomes large. We consider two cases: (i) finite k with infinite n and (ii) infinite k with finite n.

3.1 Finite k and Infinite n

With k being odd, each local sample median τ_i is an unbiased estimator of θ. For Scheme 1, the asymptotic MSE equals the finite sample expression (18). For Scheme 2, the MSE is calculated using the asymptotic distribution of a central order statistic[7, pp. 223]:

the limiting distribution of $\sqrt{n}\, f_{m:n}\left(F^{-1}_{m:n}(p)\right)\dfrac{\tau_0 - F^{-1}_{m:n}(p)}{\sqrt{p(1-p)}}$ is a standard normal, where

$m = [np]+1$, $[np]$ denotes the largest integer less than or equal to np. For sample median, $m = \dfrac{n+1}{2}$, which implies that $p = \dfrac{1}{2}$. Hence the MSE of Scheme 2 is given by

$$MSE_2(\tau_0) = \frac{1}{4n\left(f_{(k+1)/2:k}\left(F^{-1}_{(k+1)/2:k}(0.5)\right)\right)^2} \tag{23}$$

Using the above limiting distribution, the asymptotic MSE of a central estimator, which is the sample median of nk random samples, is given by

$$MSE_{\text{CEN}}(\tau_0) = \frac{1}{4nk\left(f(0)\right)^2} = \begin{cases} \dfrac{1}{nk} & \text{for Laplace} \\ \dfrac{\pi^2}{4nk} & \text{for Cauchy} \end{cases} \tag{24}$$

3.2 Infinite k and finite n

Using the limiting distribution of section 3.1, the following results are obtained in a straight forward fashion.

$$MSE_1(\tau_0) = \frac{1}{n}\mathrm{Var}(\tau_i) = \frac{1}{4nk\left(f(0)\right)^2}, \text{ same as eq. (24).}$$

$$MSE_2(\tau_0) = \begin{cases} \frac{1}{k}M & \text{for Laplace} \\ \frac{\pi^2}{4k}M & \text{for Cauchy} \end{cases} \qquad (25)$$

where M is the second moment of the $\left((n+1)/2\right)^{\text{th}}$ order statistic of a random sample of size n from a standard normal distribution.

For the central estimator eq. (24) remains valid.

4. Discussion of Numerical Results

Numerical results based on the equations developed in the previous sections are shown in Tables 1 through 8. Table 1 shows the finite sample MSE performances of Schemes 1 and 2 , and the central median estimator, for the Laplace distribution. Similar comparisons are made for the Cauchy density in Table 2. In Table 2, the entries are provided only for $n \geq 5$, because for Cauchy density, the variances of the first and the second order statistics do not exist[9]. Also shown in these tables, for comparison purposes, are the MSEs of the central BLUE. The MSE expressions for BLUE are obtained from the standard results[[7], pp. 171-173, see also [8], pp. 173]. It is observed that, whereas for the Laplace, the sample mean of the local sample medians has a smaller MSE than the sample median of the local medians, the converse is true for the Cauchy. Also, the best decentralized estimator, that is the best of Schemes 1 and 2, has a slightly larger MSE than the central median estimator. Tables 3 and 4 show the MSE of Scheme 1 corresponding to unequal sensor sample sizes, for Laplace and Cauchy densities respectively. A comparison of these tables with Tables 1 and 2 reveal that assigning different samples to different sensors, in general, does not provide any smaller MSE than that obtained with equal sample size for all the sensors(significant reduction is achieved only for the case of Cauchy with a total sample size of 25, Tables 2, 4).

The asymptotic results are shown in Tables 5 and 6, for (i) finite k and infinite n case and in Tables 7 and 8, for (ii) infinite k and finite n case. For Laplace density, Scheme 1 has a smaller MSE than Scheme 2, for any k of case (ii) and for $k \geq 9$ of case (i) (see Tables 5, 7). This is understandable, because for $k = 1$ in case(i), the sensor sample is distributed as Laplace and median is a better estimator than mean for any n. As k increases, the distribution of the median deviates from a Laplace, and for sufficiently large k, the mean of the medians has a smaller MSE than the median of the medians. For case (i) and Cauchy density, Scheme 1 has a lower MSE than Scheme 2, if k exceeds 9, whereas the converse is true for smaller k values(see Table 6). This is different from the finite sample case of Cauchy where the median of the local median has a smaller MSE than the average of the local medians. For case (ii), irrespective of the density being Laplace or Cauchy, Scheme 1 has a smaller MSE than Scheme 2(Table 8). Also, the sample mean of the local medians is equivalent to the central median estimator(see (24)). This is also understandable, because for large k, the median is asymptotically normal, and the arithmetic mean is UMVUE. Finally, as in the finite sample situation, for case (i), the central median has a smaller MSE than the better of Schemes 1 and 2, for both Laplace and Cauchy densities.

5. Conclusions

In this paper we evaluated the mean squared error performances of two decentralized estimators and a central median estimator of the location parameter of Laplace and Cauchy densities. It is well known that the median has a smaller MSE than the mean for Laplace, whereas the converse holds true for the normal. A some what surprising result is that for the Laplace density, the average of the local medians has a smaller MSE than the median of the local medians. To some extent, this behavior can be understood, if we consider the fact that the density of the median is not exactly a Laplace. Another result pertaining to the average of local median estimator is that, assigning different samples to different sensors, in general, does not provide any smaller MSE than that obtained with equal sample size for all the sensors. Finally, the central median estimator has a smaller MSE than the better of the two decentralized estimators. This confirms the expected loss in performance associated with an estimation procedure that is based on grouped data.

References

1. S.S. Iyengar, R.L. Kashyap, and R.N. Madan, "Distributed Sensor Networks-Introduction to the Special Section," *IEEE Trans. SMC.*, Sept. 1991, pp. 1027-1031.
2. C.Y. Chong, E. Tse, and S. Mori, "Distributed Estimation in Networks," *Proc. Amer. Contr. Conf.*, 1985., 1983, pp. 294-300.
3. E.B. Hall, A.E. Wessel, and G.L. Wise, "On Fusion in Estimation Theory," *Proc. Twenty-Sixth Ann. Allerton Conf. Commun., Contr., Comput.*, Monticello, IL, Sept. 1988, pp. 599-608.
4. K.C. Chang, C.Y. Chong, and Y. Bar-Shalom, "Joint Probabilistic Data Association in Distributed Sensor Networks," *Proc. Amer. Contr. Conf.*, 1985., pp. 817-822.
5. R. Viswanathan, "A Note on Distributed Estimation and Sufficiency," *IEEE Tran. IT.*, Sept. 1993, pp. 1765-1767.
6. M. Su, "Performance of Some Schemes for Distributed Location Parameter Estimation," M.S. Thesis, Department of Electrical Engineering, Southern Illinois University, Carbondale, IL-62901, Feb. 1996.
7. B.C. Arnold, N. Balakrishnan, and H.N. Nagaraja, *A First Course in Order Statistics*, New York: John Wiley, 1992.
8. N.L. Johnson, S. Kotz, and N. Balakrishnan, *Continuous Univariate Distributions*, Vol. II, 1995, New York: John Wiley.
9. V.D. Barnett, "Order Statistics Estimators of the Location of the Cauchy Distribution," *JASA*, Dec. 1966., pp. 1205-1217. Corrections, *JASA*, Mar. 1968, pp. 383-385.

k	r	n	l (sch.2)	$N = nk$	MSE_1	MSE_2	MSE_{CEN}	MSE_{BLUE}
1	1	1		1	2.000000		2.000000	
1	1	3		3	0.666667			
3	2	1	1	3	0.638889	0.638889	0.638889	0.5895
1	1	5		5	0.400000			
5	3	1	1	5	0.351181	0.351181	0.351181	
1	1	7		7	0.285714			
7	4	1	1	7	0.235648	0.235648	0.235648	0.2122
1	1	9		9	0.222222			
3	2	3	1	9		0.8474329		
3	2	3	2	9	0.212963	0.2218010	0.175068	0.1581
3	2	3	3	9		0.8474328		
1	1	15		15	0.133333			
3	2	5		15	0.127778			
5	3	3	2	15	0.117060	0.1274922	0.096349	0.088
5	3	3	3	15		0.463247		
7	4	3	2	21	0.078549	0.0879615	0.065443	
7	4	3	3	21		0.3094907		
5	3	5	2	25		0.1627839		
5	3	5	3	25	0.070236	0.0745546	0.053669	
5	3	4	4	25		0.1627840		
5	3	4	5	25		0.6778901		

Table 1 Finite Sample MSE Comparisons for Laplace Distribution

k	r	n	l (sch.2)	$N = nk$	MSE_1	MSE_2	MSE_{CEN}	MSE_{BLUE}
5	3	1		5	1.221166		1.221166	1.221
7	4	1		7	0.612303		0.612303	0.6072
				9			0.408903	0.3865
5	3	3	2	15	0.407055	0.2802118	0.205121	
7	4	3	2	21	0.204101	0.1861224	0.136366	
5	3	5	2	25		0.3627321		
5	3	5	3	25	0.244233	0.1564317		
5	3	5	4	25		0.3627321		

Table 2 Finite Sample MSE Comparisons for Cauchy Distribution

Sensor i	k_i	r_i	*n*	*N = nk*	*MSE*
1	3	1	1	3	2.680556
1	3	2	1	3	0.638889
1	3	3	1	3	2.680556
1	3	1			
2	3	2			
3	3	3	3	9	0.335715
1	3	2			
2	3	2			
3	3	3	3	9	0.212963
1	3	2			
2	5	3			
3	7	4	3	15	0.115522
1	5	3			
2	7	4			
3	9	5	3	21	0.078105
1	3	2			
2	3	2			
3	5	3			
4	5	3			
5	5	3	5	21	0.085667

Table 3 MSE of Scheme1 for Unequal Sensor Sample Sizes (Laplace Distribution)

Sensor i	k_i	r_i	*n*	*N = nk*	*MSE*
1	5	3	1	5	1.221167
1	7	4	1	7	0.612303
1	7	5	1	7	1.678037
1	7	3	1	7	1.678037
1	5	3			
2	5	3			
3	5	3	3	15	0.407056
1	7	4			
2	7	4			
3	7	4	3	21	0.204101
1	7	3			
2	7	4			
3	7	3	3	21	0.385320
1	7	4			
2	9	5			
3	9	5	3	25	0.153273
1	5	3	5		
2	5	3	5		
3	5	3	5		
4	5	3	5		
5	5	3	5	25	0.244233

Table 4 MSE of Scheme 1 for Unequal Sensor Sample Sizes (Cauchy Distribution)

k	*r*	$nMSE_1$	$nMSE_2$	$nMSE_{CEN}$
1	1	2.000000000	1.000000000	1.0000000
3	2	0.638888889	0.444444444	0.3333333
5	3	0.351180456	0.284444444	0.2000000
7	4	0.235647676	0.208979592	01428571
9	5	0.175068334	0.165119678	0.1111111
11	6	0.138280556	0.136462543	0.090909
13	7	0.113765832	0.116275777	0.076923
15	8	0.096349490	0.101289121	0.0666666
17	9	0.083383706	0.089723235	0.0588235
19	10	0.073380956	0.080527225	0.0526315

Table 5 Asymptotic MSE for Laplace Distribution Case(i) : finite *k* and infinite *n*

k	r	$nMSE_1$	$nMSE_2$	$nMSE_{CEN}$
1	1		2.467400863	2.467400863
3	2		1.096622606	0.822467204
5	3	1.221251596	0.701838468	0.493480322
7	4	0.612081463	0.515636425	0.352485945
9	5	0.408659014	0.407416435	0.274155735
11	6	0.306803520	0.336707797	0.224309237
13	7	0.245622174	0.286898952	0.189800124
15	8	0.204798415	0.249920864	0.164493441
17	9	0.175617591	0.221383188	0.145141271
19	10	0.153718970	0.198692944	0.129863243

Table 6 Asymptotic MSE for Cauchy Distribution
Case(i) : finite k and infinite n

n	$kMSE_1 = kMSE_{CEN}$	$kMSE_2$
1	1.000000000	1.000000000
3	0.333333333	0.448671108
5	0.200000000	0.286833664
7	0.1428571	0.210446863
9	0.111111111	0.166101283
11	0.090909	0.137162435
13	0.076923	0.116798996
15	0.066666667	0.101694658
17	0.0588235	0.090046582
19	0.0526315	0.080790976

Table 7 Asymptotic MSE for Laplace Distribution
Case(ii) : infinite k and finite n

n	$kMSE_1 = kMSE_{CEN}$	$kMSE_2$
1	2.467401	
3	0.822467	1.107051549
5	0.4934802	0.707733674
7	0.3524858	0.519256804
9	0.2741556	0.409838474
11	0.2243091	0.338434731
13	0.1898	0.288189961
15	0.1644934	0.250921490
17	0.1451412	0.222180284
19	0.1298632	0.199347357

Table 8 Asymptotic MSE for Cauchy Distribution
Case(ii) : infinite k and finite n

Abate, J. 923
Agrawal, R. 239
Agrawal, S.K. 325
Allen, B.M. 893
Anantharam, V. 66, 84
Anjum, F.M. 855
Appleby, B.D. 372
Balakrishnan, V. 142
Balasubramanian, R. 516
Ball, J.B. 980
Banerjee, A. 249
Barmish, B.R. 160
Barrett, G. 786
Bertsimas, D. 712
Bhattacharya, P. 566
Blake, I.F. 707
Bouthellier, P.R. 363
Boutros, J.J. 322
Braatz, R.D. 970
Brandt-Pearce, M. 180
Brassil, J. 447
Brodersen, R. 623
Brooks, C.A. 768
Burkhart, R.H. 556
Cabral, H.A. 504
Calderbank, A.R. 662
Chang, R.C.H. 806
Che, H. 22
Chen, J. 170
Chen, P.-C. 835
Cheng, J. 1009
Cheng, J.-F. 494
Chernova, N. 220
Cohen, N. 447
Coraluppi, S. 353
Costello, Jr., D.J. 504
Coyle, E.J. 903
Cruz, R.L. 10
Dahleh, M.A. 372
Das, S. 997
Deller, Jr., J.R. 382
Dhiantravan, Y. 402
Duffield, N.G. 741
Dumer, I. 683
Dumont, G.A. 980
Duursma, I.M. 322
Elmasry, G.F. 102
Ephremides, A. 44
Faiz, N. 325
Fan, H. 412
Featherstone, A.P. 970
Feng, G.-L. 286
Feron, E. 372
Fields, J. 312
Fitz, M.P. 516
Forney, Jr., G.D. 432
Foss, S. 220
Frank, C.D. 633
Frazho, A.E. 112
Frey, B.J. 482
Gadkari, S. 74, 92
Gerstel, O. 32
Ghosh, B.K. 593
Giallorenzi, T.R. 200
Glynn, P.W. 760
Gollamudi, S. 392
Gotesman, M. 815
Guess, T. 190
Harris, J.M. 865
Hjalmtysson, G. 228
Ho, S.L. 960
Ho, T.-J. 796
Holloway, L.E. 776
Holtzman, J.M. 835
Huang, Y.F. 392
Huffman, W.C. 302
Hui, J.Y. 54
Isabelle, S.H. 613
Iyer, U. 526
Jamshidi, M. 933
Jiang, H. 269
Jin, H. 469
Jordan, S. 269
Kapoor, S. 392
Klapper, A. 875
Kolodziej, C. 950
Konstantopoulos, T. 913
Kovalevskii, A. 220
Krishnakumar, A.S. 693
Krogmeier, J.V. 516
Kschischang, F.R. 482
Kumar, S.P. 865
Kwon, S.M. 210
Kwong, R.H. 806
Lafortune, S. 786
Lagoa, C.M. 160
Lehnert, J.S. 642
Li, G. 942, 960
Li, H. 942
Li, S.-Q. 22
Lin, S.-J. 913
Lin, T.M. 382
Linder, T. 62
Ling, F. 633
Lo, J.T.-H. 982, 991
Lok, T.M. 642
Lopez Benitez, N. 815
Lu, J. 122
Mandayam, N.B. 835
Marcus, S. 353
Maric, S. 703
McCanless, J.C. 200
McConley, M.W. 372
McEliece, R.J. 469, 494
Melsa, P.J.W. 546
Michail, A. 44
Min, P.S. 210
Mitra, U. 170
Moreno, O. 703
Mouftah, H. 1
Nagaraj, S. 392
Nayeri, M. 382, 526
Nowak, R.D. 506
O'Donnell, I. 623
Okino, C.M. 10
Olurotimi, O. 997
Paschalidis, I.C. 712
Pellikaan, R. 276

Peterson, C. 583
Pless, V. 312
Poor, H.V. 574, 603
Pottie, G.J. 652
Priemer, R. 402, 950
Puhalskii, A.A. 722
Rains, E.M. 662
Rajan, R. 239
Rajaram, S. 776
Ramanan, K. 732
Ramaswami, R. 32
Rao, R.R. 825
Rao, T.R.N. 286
Rohrs, C.E. 546
Rokhsaz, K. 333
Rose, K. 74, 92
Rosenthal, J. 583, 893
Rotea, M.A. 122
Roth, R.M. 673
Sain, M.K. 343
Sarkar, S. 457
Sasaki, G. 32
Sawan, M.E. 333
Shcherbakov, P.S. 160
Shen, B.-Z. 294
Sheng, S. 623
Shin, S.W. 210
Shor, P.W. 662
Shue, S.-P. 333
Sidhu, I. 269
Sinha, A.S.C. 132
Sivarajan, K.N. 457
Sloane, N.J.A. 662
Smith, M.J. 112
Spencer, Jr., B.F. 343
Stone, K. 623
Strait, J.C. 422
Su, M. 1009
Tabatabaee, V. 479
Taha, O. 980
Tarn, T.J. 593
Tarokh, V. 62, 707
Tassiulas, L. 479, 855
Teuscher, C. 623
Tsang, K.M. 942, 960
Tse, D. 259
Tsitsiklis, J.N. 712
Tzeng, K.K. 294
Ulukus, S. 845
Urbanke, R. 693
Varanasi, M.K. 190
Varvarigos, E. 249
Venkatesan, S. 66, 84
Viswanathan, R. 1009
Wang, C.C. 652
Wang, X. 603
Wang, Y. 566
Weiner, P.A. 583
Weiss, A. 732
Whitt, W. 722, 923
Williamson, G.A. 536
Wilson, S.G. 200
Won, C.-H. 343
Wong, T.F. 642
Wornell, G.W. 613
Wu, X. 286
Xi, N. 593
Xiao, D. 593
Xiao, Q. 903
Yang, O.W. 1
Yates, R.D. 845
Yaz, E.E. 152
Yaz, Y.I. 152
Yiin, L.-H. 574
York, E.V. 885
Younce, R.C. 546
Yu, Z. 469
Zajic, T. 751
Zeger, K. 62
Zhang, H. 1
Zimmerman, S. 536
Zorzi, M. 825

UNIVERSITY OF ILLINOIS-URBANA
3 0112 033279776